



عمادة البحث العلمي

أمن الشبكات والنظم
المفاهيم والتقنيات

إعداد

الدكتور
حسن الصلاي

الدكتور
خالد الشلفان



٢٠١٣م - ١٤٣٥هـ

المملكة العربية السعودية
وزارة التعليم العالي
جامعة الإمام محمد بن سعود الإسلامية
عمادة البحث العلمي



أمن الشبكات و النظم

المفاهيم و التقنيات

إعداد

د. حسن مبروك الصلاي   د. خالد بن عبد العزيز الشلفان

١٤٣٥هـ- ٢٠١٣م

ح

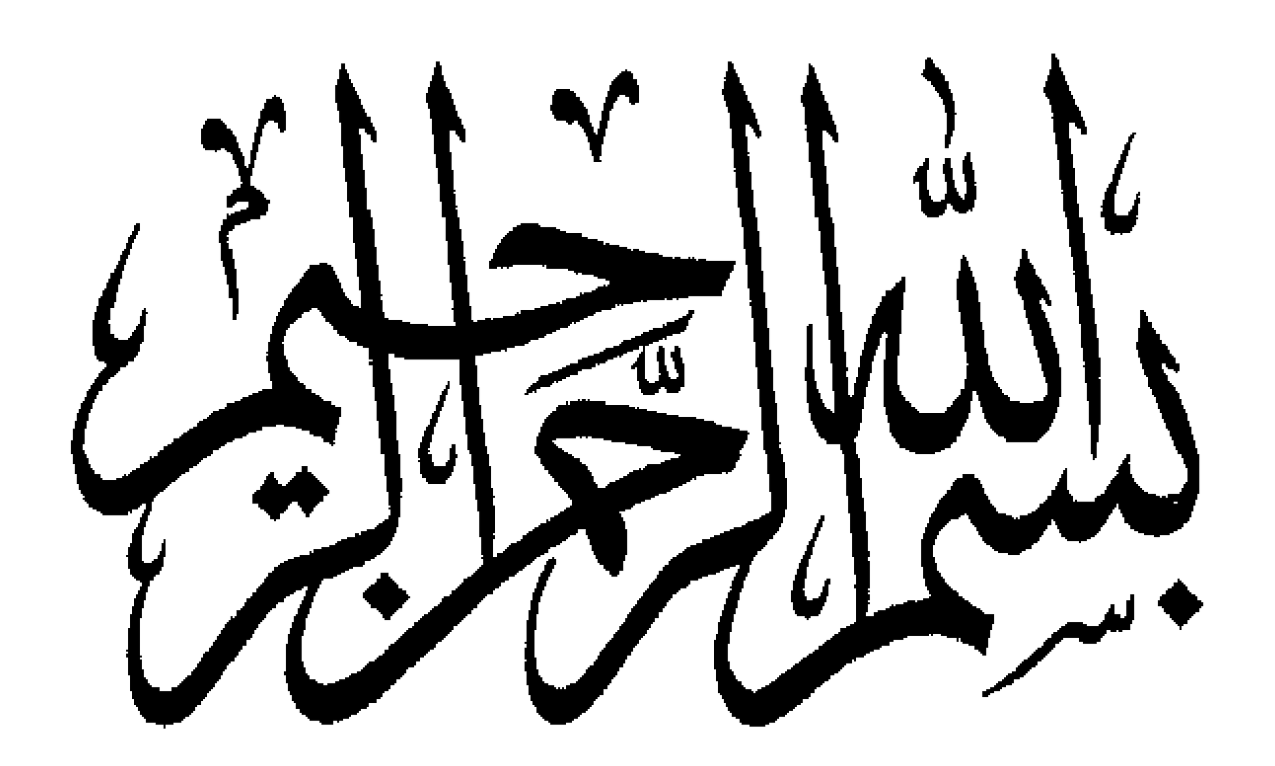
جامعة الإمام محمد بن سعود الإسلامية، ١٤٣٤هـ
فهرسة مكتبة الملك فهد الوطنية أثناء النشر

الصلاي، حسن مبروك
أمن الشبكات و النظم المفاهيم و التقنيات./
حسن مبروك الصلاي، خالد بن عبد العزيز الشلفان
– الرياض، ١٤٣٥هـ
٢٦٤ص، ١٧×٢٤سم

ردمك: ٢-٢٠٩-٥٠٥-٦٠٣-٩٧٨

١- أمن المعلومات ٢- شبكات المعلومات
أ.الشلفان،خالد بن عبدالعزيز(مؤلف مشارك)
ب. العنوان
ديوي ٠٠٥,٨ ١٤٣٥/١٣٠٦

رقم الإيداع ١٤٣٥/١٣٠٦
ردمك: ٢-٢٠٩-٥٠٥-٦٠٣-٩٧٨


بسم الله الرحمن الرحيم

الطبعة الأولى
١٤٣٥هـ - ٢٠١٣م

تقديم عميد البحث العلمي

الحمد لله رب العالمين، والصلاة والسلام على أشرف الأنبياء والمرسلين، وعلى آله، وصحبه أجمعين، ومن تبعهم بإحسان إلى يوم الدين . أما بعد:

فقد نصت المادة الأولى في نظام مجلس التعليم العالي والجامعات في المملكة العربية السعودية على أن الجامعات السعودية مؤسسات علمية وثقافية، تعمل على هدي الشريعة الإسلامية وتقوم بتنفيذ السياسة التعليمية بتوفير التعليم الجامعي والدراسات العليا، والنهوض بالبحث العلمي، والقيام بالتأليف، والترجمة، والنشر وخدمة المجتمع في نطاق اختصاصها.

وعمادة البحث العلمي بجامعة الإمام محمد بن سعود الإسلامية في سبيل تحقيق أهدافها المنوطة بها تعنى بنشر البحوث العلمية، والرسائل الجامعية، وترجمة ما ترى فيه النفع إلى العديد من اللغات العالمية، وتستكتب في السلاسل الثقافية التي تصدرها العديد من المتخصصين؛ لتقدم المتميز من الأعمال العلمية.

وها هي تضع بين يدي القراء هذا البحث العلمي الذي وافق المجلس العلمي في الجامعة على نشره بقراره ذي الرقم (٣١٨- ١٤٣٣هـ/١٤٣٤هـ في جلسته (التاسعة عشرة) المعقودة في ٨/ ٧ / ١٤٣٤هـ، والموسوم بـ (أمن الشبكات و النظم: المفاهيم و التقنيات) الذي أعده كل من الدكتور: حسن بن مبروك الصلاي و الدكتور خالد بن عبد العزيز الشلفان، نسأل الله ـ عز وجل ـ أن ينفع بهذا البحث، إنه سميع مجيب.

عميد البحث العلمي

د. عبد الرحمن بن عبد العزيز المقبل

مُقَدِّمَةُ الْكِتَاب

إن الحاسوب ملأ الدنيا وشغل الناس. هو بحق عصب الحياة المعاصرة، إذ ترتكز عليه اليوم أغلب الأنشطة، ويمس مختلف الجوانب الاقتصادية، والسياسية، والاجتماعية، ناهيك عن الجوانب البحثية، والمعرفية. ومع انتشار الحواسيب الشخصية، وتقارب عالَمَي الحواسيب والاتصالات، وظهور أجيال من الجوالات الكفية، وظهور الشبكات اللاسلكية، والحواسيب المتنقلة وتطبيقاتها، لترتبط في نسيج شبكي معقد ومفتوح هو شبكة الانترنت، ذلك الفضاء الواسع للمعلومات، والمبادلات التجارية، والأنشطة المالية، والتأثير الإعلامي، والتوجيه السياسي الاقتصادي. يوفر هذا الفضاءُ جملةً من الخدمات الرقمية، لعل الأشهر منها اليوم خدمة الويب، التي جعلت العالم اليوم كالقرية الصغيرة، وقرَّبَتْ البعيد، وقاربت بين الأسواق، وتوجهت نحوها المؤسسات والشركات، حتى الحكومات لديها بما يُعرَفُ بـ(منظومات الحكومات الالكترونية) اليوم. هذا الفضاء الرحب – هو في الوقت ذاته – مجال خصب أيضًا للجرائم الالكترونية، والاختراقات الحاسوبية، والأعمال الإرهابية، وغيرها، إذ بقدر ما أصبح تبادل المعلومة أسرع وأدق وأكثر مرونة، أصبحت الهجومات الحاسوبية أسرق وأدق في إصابة الهدف، وأكثر مرونة في تنفيذ الجريمة. ولا أدل على ذلك من حادثة (دودة سلامر) التي اكتسحت آلاف الحواسيب في زمن قصير جدًا، وخلَّفَتْ أضرارًا بالغةً في مُختلفِ أنحاء العالم، حتى أنها طالت بعض أجهزة مراقبة المنشآت النووية، مما يوضح حجم الأضرار التي يمكن أن تخلفها مثل هذه الهجومات. وبما أن الحفاظ على المعلومات، والدفاع عن النفس، وتوقي الهجومات، حق مشروع، وضرورة ملحة نَتَجَ (**علم أمن الحاسوب**)، الذي يدور حول سبل وطرق الوقاية، واكتشاف العمليات غير المسموح بها على حاسوب بعينه، بناء على سياسة أمنية معينة، تحدد مَنْ ومَاذَا ومَتَى وكيفَ يمكن استعمال موارد ذلك الحاسوب، ويعرف أيضا بـ(**أمن النظم**)، لأن نظم التشغيل هي الأساس في إدارة موارد الحاسوب. كما ظهر علم: (**أمن الشبكات**) الذي يدور أساسا على كيفية حماية المعلومة، والتأكد من الحفاظ على سريتها وسلامتها عند تبادلها على الشبكة، والطرق الآمنة للتأكد من هوية المتخاطبين على الشبكة. وأعم منه (**علم أمن المعلومات**)، إذ يدور على حماية المعلومة بغض النظر على مكان وجودها – حاسوبيًا كان أو ورقًا –. ونحن نعرض في هذا الكتاب مقدمة للمفاهيم الأساسية، ومبادئ أمن الشبكات والنظم.

الْفَصْلُ الأَوَّل
مُقَدِّمَاتٌ عَامَّة

يَهْدِفُ هَذَا الْفَصْلُ إِلَى:

1. التعريف بأهداف وهجومات وخدمات أمن المعلومات.
2. التعريف بالآليات الأمنية، التي توفر خدمات أمن المعلومات.
3. تقديم أهم تقنيات تنفيذ هذه الآليات.

1- مُقَدِّمَةُ الْفَصْلِ

أصبحت المعلومات من أهم الموارد والأصول التي تقوم عليها حياة الأفراد والمؤسسات، بل والمجتمعات اليوم. ويتطلب منا عصر المعلومات هذا الذي نعيش فيه الاحتفاظَ بكثير من المعلومات التي تمس أغلب الأنشطة الإنسانية اليوم، فلقد أصبحت المعلومة قطب الرحى في كثير من المجالات المعرفية، والإقتصادية، والسياسية، والإجتماعية، وغيرها، مما يجعل الحفاظَ عليها من أهم المسائل في ظل التعقيدات التقنية لهذا العصر الرقمي. وفي ظل هذه المتاهة من الشبكات والخدمات – ذات الدرجات المختلفة من الأمن (أو انعدام الأمن) – فإنه لا بد من التأكد من سرية المعلومة حتى لا يطلع عليها إلا من هو مخول له بذلك، وسلامتها من كل تغيير، وتوفرها عند الحاجة إليها، مع تحديد المسؤوليات للمحاسبة القانونية لتعديات المستخدمين. ولْنَعُدْ للقصة من أولها.

قبل عقود قليلة كانت المعلومات تخزن في شكلها الورقي حيث لا يصل إليها إلا فئة محددة من الأشخاص المخول لهم، إما بالإطلاع أو تغيير الملفات الورقية، وبهذا تضمن سرية وسلامة المعلومات، وأما توفرها فيكفي لضمانها تحديد شخص على الأقل له صلاحية الوصول للملفات الورقية في كل الأوقات. ومع بزوغ فجر عصر تقنية المعلومات بظهور جيل من أجهزة الحاسب الضخمة، تتبادل البيانات فيما بينها عن طريق الأشرطة الممغنطة، التي تحوي المعلومات بشكلها الرقمي، أصبح أمن المعلومات يتركز على حماية الولوج إلى غرف هذه الحواسيب لحماية البيانات الرقمية والعتاد المادي من الإتلاف والسرقة.

وخلال الستينيات من القرن الماضي، ومع حمى الصراع العسكري في الحرب الباردة، ظهرت الحاجة الماسة لإيجاد طرق تبادل للمعلومات بطريقة أكثر جدوى ومرونة، ومن ثم أنشأت شبكةٌ لربط الحواسيب عُرِفَت باسم: (أربنت) التي كونت فيما بعد نواة شبكة الإنترنت الحالية.

ومع ظهور الشبكات ظهرت تحديات جديدة في أمن المعلومات، وبدأت الهجمات على النظم الحاسوبية تصبح أكثر شيوعا وسهولة. وفي منتصف (1980) وقعت واحدة من أكبر الهجمات الأمنية أضرت بأكثر من 6000 حاسوب، العدد الذي يمثل قرابة عشر العدد الكلي للحواسيب في العالم في ذلك الوقت. وعلى الرغم من ظهور إصدارات لأنظمة تشغيل – أهمها نظام يونيكس – تمتلك طرق حماية أمنية ككلمات السر والتوثيق،

وتحديد صلاحيات المستخدمين، على مستويات متعددة قلّلَتْ من بعض المخاطر، ولكن لم تجعل هذه الأنظمة بمنأى عن الهجوم كما في واقعة (فيروس الشيطان) الشهيرة لعام (1996) على نظام يونكس.

ومع زيادة استخدام الحاسوب الشخصي في مكاتب العمل والمنازل، وتعدد الشبكات ذوات النطاق المحلي والواسع، وظهور شبكة النسيج العنكبوتية وخدماتها، وانتشار التجارة الإلكترونية، تزايدت نقاط الضعف والهشاشة الأمنية لكثير من الأنظمة والخدمات، مقابل انتشار وتيسر الكثير من برامج الاختراق بواجهات رسومية سهلة وتفاعلية، لا تتطلب خبرات تقنية عالية، أضحت مسألة الأمن أكثر أهمية من أي وقت مضى.

وعلى الرغم من أن متطلبات أمن المعلومات من السرية والسلامة والتوفرية، وتحديد المسؤوليات للمحاسبة القانونية، تظل كما هي من حيث إنها متطلبات إلا أنه انضاف إلى صلبها أبعاد أخرى. فلم يعد يقتصر الأمر على تحقيق هذه المتطلبات بالنسبة للمعلومات المخزنة في أجهزة الحاسب فقط، بل لا بد من الحفاظ على سرية المعلومات عند تبادلها عن طريق الشبكة، وضمان عدم تغييرها، وكذلك منع الوصول إلى المعلومة عن بُعْدٍ لغير المخولين، والحيلولة دون تعطيل الوصول إليها، والمساس بتوفريتها، وكذلك التعرف على المعتدين فيما يعرف بالتحقيق الجنائي الرقمي.

في هذا الفصل من الكتاب سنتباحث في الأهداف الثلاثة الرئيسة لأمن المعلومات وهي: السرية، والسلامة، والتوفرية[1]، ونتعرض لكيفية تهديد الهجومات الأمنية لهذه الأهداف، وعلاقة الخدمات الأمنية بها، وأخيرًا سنتعرف على آليات توفير هذه الخدمات الأمنية، ونتطرق لتقنيات تنفيذ هذه الآليات.

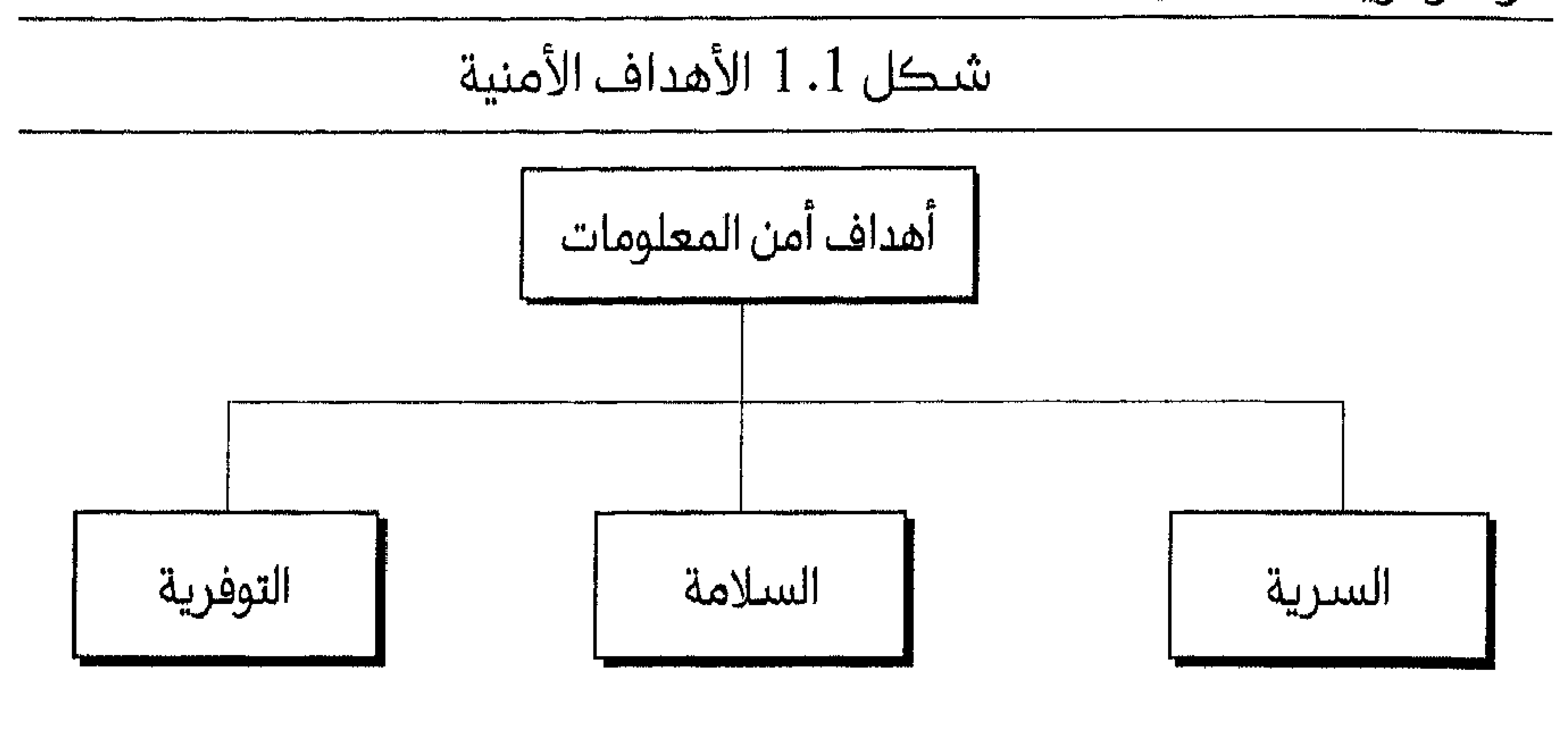
---

[1] وأما تحديد المسؤوليات فسنتعرض له باختصار في آخر فصول الكتاب عند التحدث عن سياسات أمن المعلومات وذلك أن موضوع الكتاب هو أمن الشبكات خاصة لا أمن المعلومات عامة.

2- الأهداف الأمنية

هناك ثلاث أهداف رئيسة لأمن المعلومات وهي السرية والسلامة والتوفرية (الشكل 1.1)

شكل 1.1 الأهداف الأمنية



3- السرية

تحقيق سرية المعلومات هو أبرز أهداف أمن المعلومات. ففي المجال العسكري ضمان سرية المعلومة الحساسة يعتبر الهم الأكبر فيه، وكذا بالنسبة للمؤسسات الصناعية، فإن إخفاء بعض المعلومات عن المنافسين يعتبر هدفا حيويا واستراتيجيا في ديمومة نجاحاتها، وأما بالنسبة للبنوك فإنها أيضا في حاجة إلى أن تضمن سرية معلومات حسابات عملائها.

السرية لا تختص بتخزين المعلومة، بل تشمل تبادل ونقل المعلومة على الشبكة. فعندما نرسل أو نجلب معلومة من جهاز حاسب عن بُعْدٍ فنحن بحاجة للحيلولة دون كشفها لغير مخول بالاطلاع عليها.

4- السلامة

إن المعلومة بحاجة إلى أن تتغير باستمرار، ففي النشاط المصرفي مثلا عندما يقوم عميل ما بسحب أو تحويل مبلغ مالي فإن رصيده بحاجة إلى تحديث. فسلامة المعلومة تعني أن التغييرات لا تكون إلا لصاحِبِي صلاحية التغيير، وأيضًا من خلال آليات مصرح بها. ويجدر أن يذكر أيضًا أن الإخلال بسلامة المعلومة ليس دائمًا يكون جراء اعتداء عليها، بل إن بعض أعطال النظام يمكن أن تخل بسلامة المعلومة، وتحدث فيها تغييرا غير مرغوب فيه.

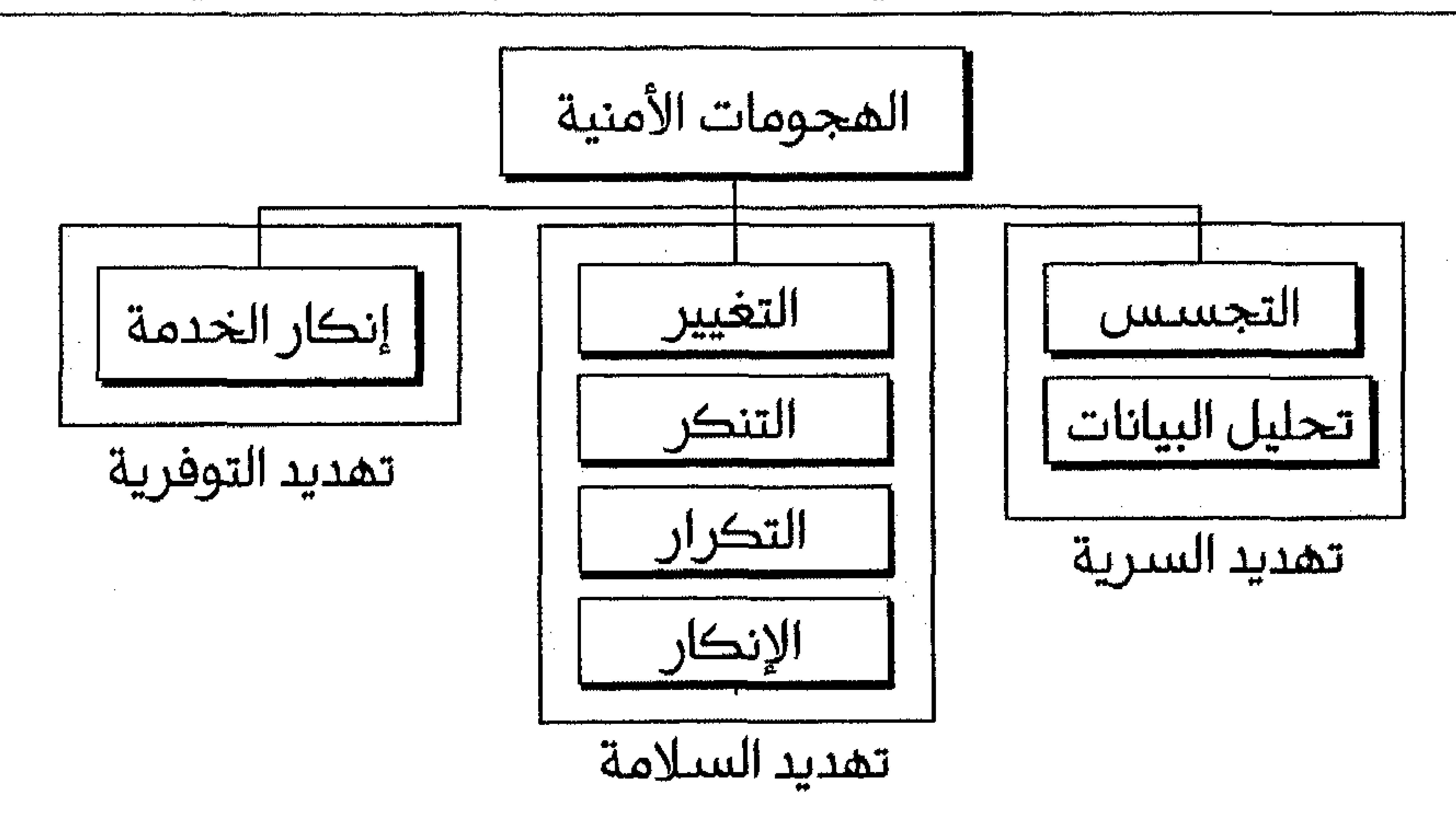
5- التوفرية

إن المعلومات إن لم تتوفر وقت الحاجة إليها فإنها تصبح في حكم المعدوم، وغير ذات جدوى. ولأن المعلومة تخضع لتغير مستمر فلا بد من أن تكون متوفرة للمخول لهم بالتغيير، فعدمُ توفريةِ المعلومة يَجُرُّ ضررًا شديدًا على المؤسسة كضرر عدم سرية المعلومة أو سلامتها، ولك أن تتصور ما يحدث لو أن عملاء مصرف ما لا يتمكنون من إجراء عملياتهم المصرفية، وأثر ذلك على المؤسسة المصرفية.

6- الهجومات الأمنية

يمكن أن نصنف الهجومات الأمنية إلى ثلاثة أصناف؛ تبعا للهدف الأمني الذي يخل به، وفيما بعد سنصنفها إلى صنفين أكثر شمولا بالنظر إلى تأثيرها على النظام المستهدف. الشكل 2.1 يعرض التصنيف الأول.


شكل 2.1 تصنيف الهجمات حسب الأهداف الأمنية



6.1 هجومات الإخلال بالسرية

بصفة عامة هناك نوعان من الهجومات تخل بالسرية وهي: التجسس وتحليل البيانات. فالتجسس يرجع إلى اطلاع المتطفل (وسنرمز إليه ب **"الدخيل"**) على الاتصال الشبكي بين مرسل (وسنرمز إليه ب **"زيد"**) ومستقبل (وسنرمز إليه ب **"عبيد"**) بتعرضه للبيانات أثناء انتقالها على الشبكة لاستعمالها لصالحه. وللحيلولة دون ذلك فإننا نستعمل تقنيات التشفير التي سنشرحها قادمًا في هذا الكتاب، وبالرغم من استعمال هذه التقنيات فإن الدخيل يمكن أن يحصل على نوع آخر من المعلومات من

خلال مراقبته للاتصال، وتحليل البيانات التي تتدفق فيه. فمثلا يمكن أن يحصل الدخيل على عنوان البريد الالكتروني لزيد أو عبيد، كما يمكن أن يحصل على طلب من زيد، وجواب طلبه من عبيد، يساعده على التعرف على نوعية العملية الإجرائية بينهما.

6.2 هجومات الإخلال بالسلامة

إن هجومات الإخلال بالسلامة هي الأكثر تعددا. فهجومات التغيير تحدث بعد التعرض، والتقاط البيانات من طرف الدخيل ليحدث فيها التغيير اللازم الذي يجعلها في صالحه، كما لو أجرى زيد عملية مصرفية مع عبيد فإن الدخيل سيحول نوع العملية لصالحه بإحداث التغييرات اللازمة. كما يلجأ الدخيل أحيانا إلى حذف أو تأخير وصول المعلومة للضرر بزيد أو عبيد، أو للانتفاع منهما.

وعندما ينتحل الدخيل شخصية زيد أو عبيد فإن هذا الهجوم يعرف بالتنكر، كما لو يحصل الدخيل على كلمة سر الحساب المصرفي بادعائه لعبيد أنه زيدٌ أو بالعكس، يدعي لزيد أنه عبيد الذي يمثل صرافه.

أما أن يحتفظ الدخيل بنسخة من عملية مصرفية أجراها زيد مع عبيد ثم يقوم بعدَ حِينٍ بإعادة تنفيذ نفس هذه العملية باستعمال نسختها، فهذا ما يعرف بهجوم تكرار العمليات باستعمال نسخها. فمثلا لو أن الدخيل تقاضى مبلغًا ماليًا جزاء عمل قام به من زيد، وتمت عملية التقاضي من خلال تحويل مصرفي فإن الدخيل لو تمكن من الاحتفاظ بنسخة من هذه العملية لأمكنه فيما بعد إعادة تنفيذها بارسالها مرة ثانية لمصرف زيد، والحصول بذلك على نفس المبلغ مرة ثانية.

أما أن ينكر زيد أو عبيد إرسال شيء للآخر أو استقباله منه فهذا ما يعرف بهجوم الإنكار. فمثلا يمكن أن ينكر زيد أنه أمر عبيدًا بتحويل مبلغ ما لشخص آخر مع أمره إياه فعلاً بذلك، وكذلك يمكن أن ينكر عبيدٌ أن زيدًا سدّدَ ما عليه من رسوم مع أن زيدًا قد سدّد بالفعل. وهذا الهجوم مختلف عما سبق لعدم دخول أي دخيل فيه.

6.3 هجومات الإخلال بالتوفرية

سنشير هنا إلى الهجوم الأكثر شيوعًا، وهو هجوم إنكار أو جحود الخدمة. يمكن للمهاجم أن يستعمل جملة متعددة من الاستراتيجيات

للقيام بمثل هذا الهجوم. فمثلاً يمكن أن يغرق المهاجم عبيدًا بطلب مزيف تلو آخر فيشغله بمعالجة الطلبات المزيفة، وتفويت معالجة الطلبات الحقيقية عليه حتى يصل الأمر بعبيد إلى رفض الطلبات الحقيقية؛ لشدة انشغاله بالطلبات المزيفة، وتتعطل بذلك خدمات عبيد فتتأثر بذلك سمعة عبيد عند عملائه، كما يمكن للدخيل أن يسحب ويحذف إجابات عبيد على طلبات زيد فيعتقد زيد أن عبيدًا معطلاً، كما يمكن له أن يحذف طلبات زيد ويحول دون وصولها لعبيد فيسعى زيدٌ في كلّ مرةٍ لإعادة طلبه، وتغرق الشبكة بطلبات زيد الذي لن يحقق مراده من عبيد.

6.4 الهجومات الخاملة والنشطة

يمكن أن نصنف الهجومات إلى نوعين آخرين أكثر شمولا وهما: الهجومات الخاملة والنشطة.

جدول 1.1 تصنيف الهجمات الخاملة والنشطة

الهجوم	خامل/ نشط	تخل ب
التجسس تحليل البيانات	خامل	السرية
التغيير التنكر التكرار الإنكار	نشط	السلامة
إنكار الخدمة	نشط	التوفرية

في الهجومات الخاملة، يسعى المهاجم للحصول على المعلومات السرية لزيد أو عبيد فقط دون المساس بها، ولكن يمكن إلحاق الضرر بهما من جهة أن المعلومات أصبحت مكشوفة، لا من جهة تغيير هذه المعلومات أو إحداث ضرر على نظاميهما، ولهذا السبب فإن إكتشاف مثل هذا النوع من الهجومات هو من الصعوبة بمكان إلا أن يشعر زيد أو عبيد بتسرب معلومات بطريق أو بآخر كنشر الدخيل لها مثلا. هذا النوع من الهجوم يمكن الحيلولة دونه باستعمال تقنيات التشفير.

أما في الهجومات النشطة فإنه يمكن إلحاق الضرر بالمعلومات والنظام كتلك الهجومات التي تخل بالسلامة والتوفرية، واكتشاف

الهجومات النشطة أسهل من الحيلولة دونها لتعدد أساليب هذه الهجومات، وتغير استراتيجياتها.

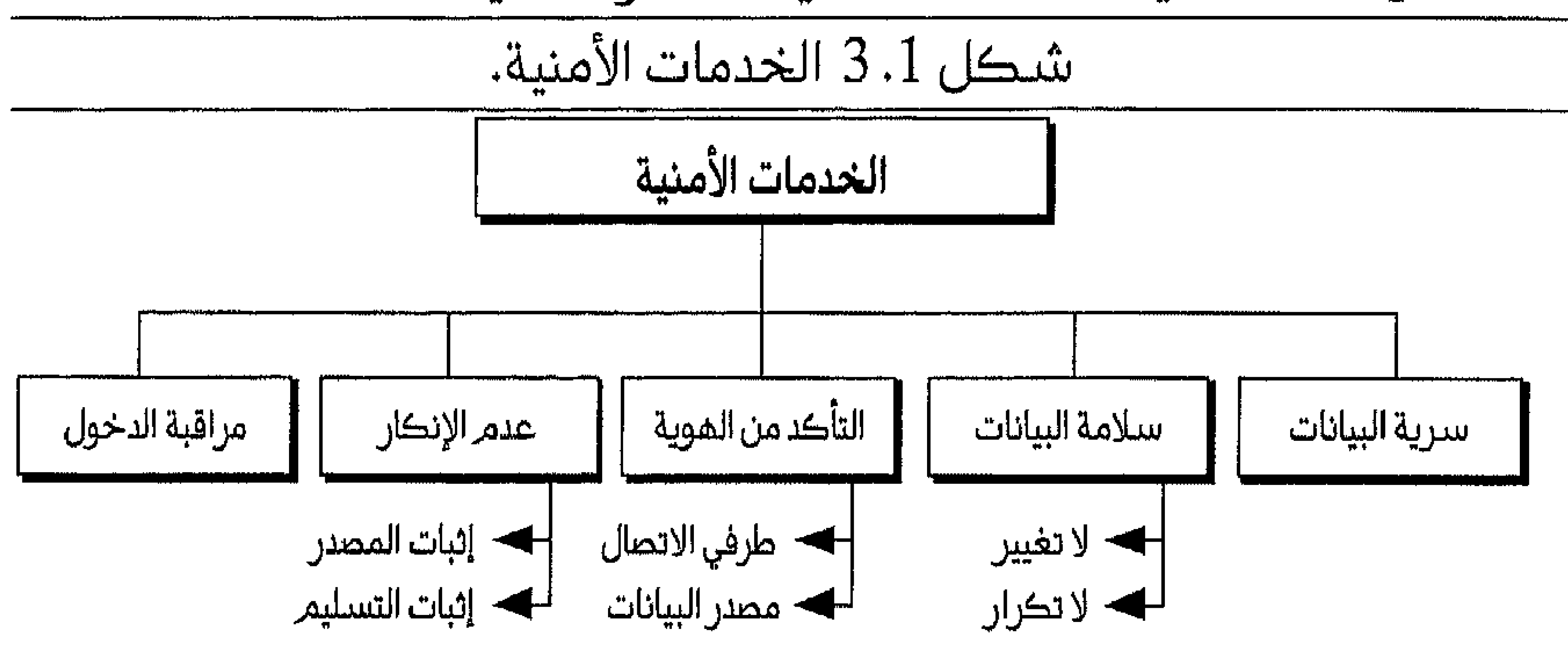
7- الخدمات والآليات الأمنية

نص الإتحاد الدولي للاتصالات في هيئة تقييسه على جملة من الخدمات الأمنية وآليات تنفيذها، فكل خدمة أمنية تنفذ عن طريق آلية أو جملة من الآليات، كما يمكن لآلية معينة أن تدمج في تنفيذ أكثرَ من خدمة أمنية. سنتعرض إلى هذه الآليات في هذا المقام بشيء من الاختصار، ونترك التفاصيل للفصول المقبلة – إن شاء الله –.

7.1 الخدمات الامنية

حَدَّدَتْ وثيقةُ x.800 الصادرةُ من الاتحاد الدولي للاتصالات الخدمات الأمنية المتعلقة بالأهداف والهجومات الأمنية السابق ذكرها آنفا. يعرض الشكل 3.1 تصنيف الخدمات الأمنية المتعارف عليها.

شكل 3.1 الخدمات الأمنية.



يلاحظ أنه من السهل أن تربط واحدة أو أكثر من هذه الخدمات بهدف أو أكثر من الأهداف الأمنية، كما يبدو واضحا أن هذه الخدمات طورت لمنع الهجمات الأمنية السالفة الذكر، فخدمةُ سرية البيانات طورت لِصَدِّ هجومات هَتْكِ سرية المعلومات. وهذه الخدمة كما نصَّتْ عليها وثيقةُ x.800 تشمل سرية البيانات بالكامل أو جزءًا منها ضد التجسس، وأيضًا الحماية ضد هجومات تحليل البيانات.

أما خدمة سلامة المعلومات فقد صُمِّمَتْ لحماية البيانات من التغيير، والتبديل، والإضافة، والحذف، وتكرار تنفيذ العمليات. ويمكن أن تشمل كل البيانات أو تقتصر على بعضها. وخدمة التأكد من الهوية في الاتصال

التزامني تضمن التأكد من هوية طَرَفَي الاتصال، وإن لم يكن الاتصال تزامنيًا تَضمّنَ التأكد من مصدر البيانات.

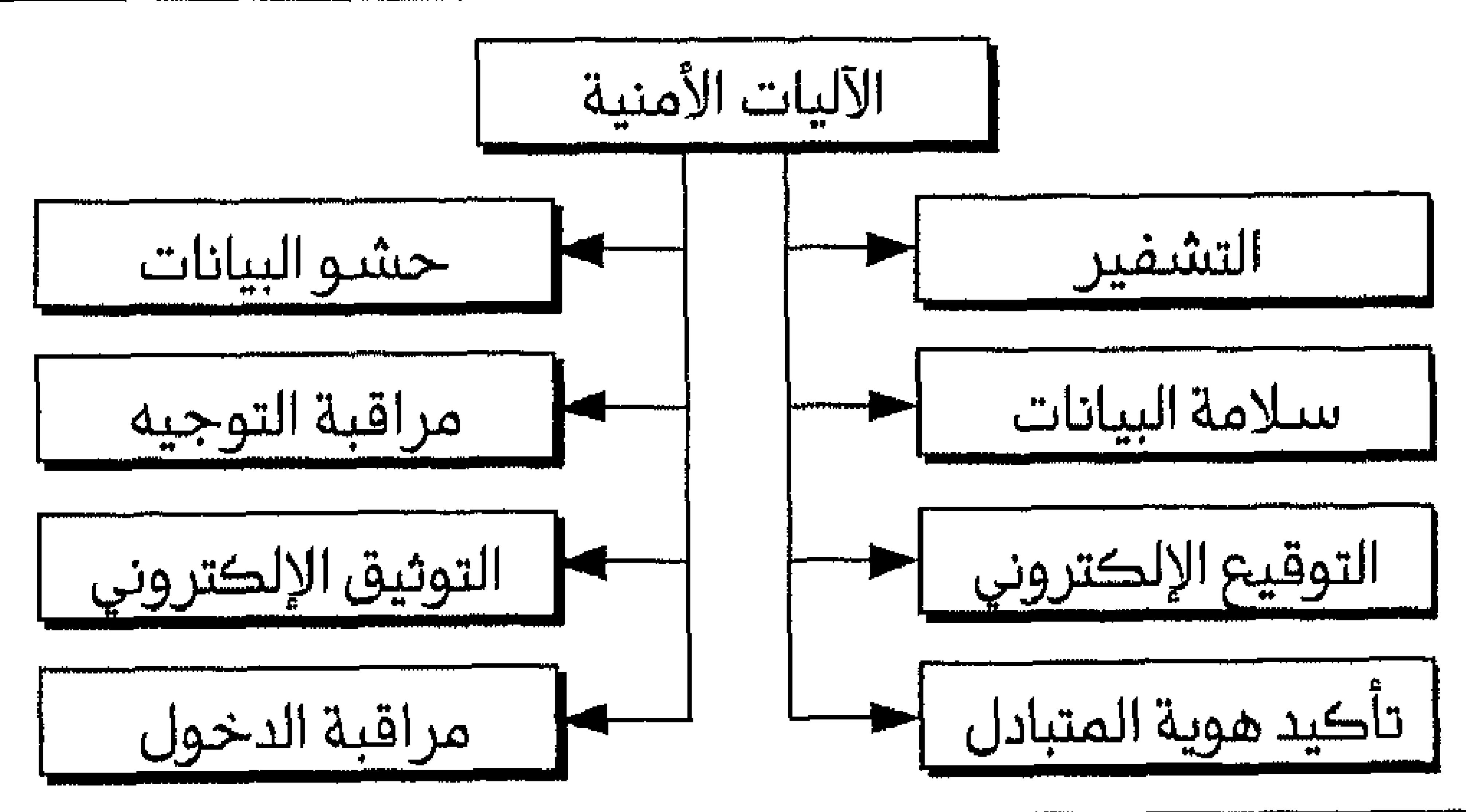
وأما خدمة عدم الإنكار ففي إثبات المصدر، فإن المستقبل يضمن إثبات هوية المرسل إليه، وفي إثبات التسليم يضمن المرسل إثبات استلام المرسل إليه للبيانات.

وأخيرًا فإن خدمة مراقبة الدخول تَحُولُ دُونَ وُلوج غير المصرح لهم للنظام. ومصطلح الدخول نقصدُ معناه الواسعَ كقراءةٍ، أو كتابةٍ، أو تغيير، أو تنفيذ للبرامج، وغير ذلك.

7.2 الآليات الأمنية

تنصح الوثيقةُ x.800 ببعض الآليات الأمنية لتوفير الخدمات الأمنية الآنفة الذكر. الشكل 4.1 يعرض تصنيفًا لهذه الآليات.

شكل 4.1 الآليات الأمنية.



فآلية التشفير تضمن سرية البيانات، سواء كان ذلك بتقنية التعمية، أو التورية، كما يمكن أن تساند غيرها من الآليات في تنفيذ الخدمات الأمنية الأخرى.

وآلية السلامة تضيف للبيانات قيمة قصيرة خاصة تستخرج من خلال تطبيق خَوَارزْمِيّةٍ معينة على البيانات نفسها. ترسل هذه القيمة مع البيانات الأصلية للمرسل إليه، وللتثبت من سلامة البيانات، فإن هذا الأخير يطبق نفس الْخَوَارِزْمِيّة على البيانات فإنْ حصل نفس القيمة المرسلة إليه فإن البيانات سليمة، وإلا فلا.

أما آلية التوقيع الالكتروني، فإنها تمكن المرسل أن يقوم بتوقيع إلكتروني على البيانات التي يرسلها للمرسل إليه، الذي يمكن له بدوره أن يتأكد من صحة هذا التوقيع. وهذه الآلية يمكن تنفيذها من خلال نظام

المفتاح الخاص والعام، الذي سنفصله في فصل مستقل من هذا الكتاب. أما في آلية تأكيد الهوية المتبادل فإنه يمكن لطَرَفَي الاتصال أن يثبت كلٌ واحدٍ للآخر أنه يعلم السر الذي من المفترض أن لا يعلمه أحد سواهما.

وأما آلية حشو البيانات فتهدف للحيلولة دون هجمات تحليل البيانات، وذلك بزرع بيانات وهمية في البيانات الأصلية. وأما تغيير مسارات إرسال البيانات بين المرسل والمستقبل فيهدف للحيلولة دون التنصت على مسار بعينه، وهذه الآلية تعرف بمراقبة توجيه البيانات.

وأما آلية التوثيق الإلكتروني فإنها تهدف لتنفيذ خدمة عدم الإنكار وذلك بإشهاد طرف ثالث موثوق عند طَرَفَي الاتِّصال يوثق عمليات تبادل البيانات بينهما لاستعمالها فيما بعد في إثبات الاستقبال أو الإرسال إن أنكر أحد طَرَفَي الاتصال عدمَ فعلِهِ لذلك.

وأخيرا فإن آلية مراقبة الدخول، تستعمل جملة من الطرق، ككلمات السر مثلا، للتثبت من أن مستخدم ما يمتلك الصلاحيات الكافية التي تمكنه من الدخول على البيانات، أو استعمال موارد معينة في النظام.

الجدول 2.1 يظهر العلاقة بين الخدمات والآليات الأمنية.

جدول 2.1 علاقات الآليات بالخدمات الأمنية.

الخدمات	الآليات
سرية البيانات	التشفير ومراقبة التوجيه.
سلامة البيانات	التشفير والتوقيع الالكتروني وآلية سلامة البيانات.
التأكد من الهوية	التشفير والتوقيع الالكتروني وتأكيد الهوية المتبادل.
عدم الإنكار	التوقيع الالكتروني وآلية سلامة البيانات وتوثيق البيانات.
مراقبة الدخول	آلية مراقبة الدخول.

8 مراجع إضافية

8.1 كتب

هناك العديد من الكتب التي تناقش الأهداف، والآليات، والهجومات الأمنية. ننصح بالرجوع إلى الكتابين التاليين:

- Bishop, M. **Computer Security**. Reading, MA: *Addison-Wesley*, 2005
- Stallings, W. **Cryptography and Network Security**. Upper Saddle River, NJ: *Prentice Hall*, 2006

8.2 مواقع

- http://www.faqs.org/rfcs/rfc2828.html
- http://fag.grm.hia.no/IKT7000/litteratur/paper/x800.pdf

9 أهم مصطلحات الفصل

Access control	مراقبة الدخول
Active attack	الهجومات النشطة
Asymmetric-key encipherment	التشفير غير التناظري
Authentication	التأكد من الهوية
Authentication exchange	تبادل التأكد من الهوية
Availability	التوفرية
Confidentiality	السرية
Cryptography	علم التشفير
Data confidentiality	سرية البيانات
Decryption	فك التشفير
Denial of service	تعطيل الخدمة
Digital signature	التوقيع الالكتروني
Encipherment	تشفير
Encryption	تشفير
Hashing	المزج
Integrity	السلامة
International Telecommunication Union (ITU-T)	الاتحاد العالمي للاتصالات
Masquerading	التنكر
Modification	التغيير
Nonrepudiation	عدم الإنكار
Notarization	التوثيق
Passive attack	الهجومات الخاملة
Private key	المفتاح الخاص
Public key	المفتاح العام
Replaying	تكرار التنفيذ
Repudiation	الإنكار
Routing control	مراقبة التوجيه
Secret key	المفتاح السري
Security attack	الهجومات الأمنية

Security goals	الأهداف الأمنية
Security mechanisms	الآليات الأمنية
Snooping	التجسس
Symmetric-key encipherment	التشفير التناظري
Traffic analysis	تحليل البيانات
Traffic padding	حشو البيانات

10 ملخص الفصل

قمنا في هذا الفصل بـ:

1. تعريف أهداف أمن المعلومات، وهي: السرية، والسلامة، والتوفرية.
2. تحديد أنواع الهجومات الأمنية منها نوعان من الهجومات التي تخل بالسرية وهما: التجسس، وتحليل البيانات. وأربعة تخل بالسلامة وهي: التغيير، والتنكر، وتكرار التنفيذ والإنكار. وواحد خاص بالتوفرية ألا وهو إنكار الخدمة.
3. تقديم خمس خدمات أمنية يمكن تحقيقها من خلال ثمان آليات تنفيذية حددها الاتحاد الدولي للاتصالات.

11 تمارين الفصل

1. عرف أهداف الأمن الثلاثة.
2. فرق بين الهجومات الخاملة والنشطة، ومثل لكل منهما بأمثلة.
3. ما الخدمات الأمنية التي ذكرت في هذا الفصل؟
4. عرف الآليات الأمنية التي نُوْقِشَتْ في هذا الفصل.
5. أي الخدمات الأمنية المضمونة عند استعمالنا لإحدى هذه الطرق لإرسال بريد لمكتب البريد:
   أ. رسالة إعتيادية.
   ب. رسالة اعتيادية مع تأكيد الاستلام.
   ت. رسالة اعتيادية مع تأكيد الوصول وتوقيع المرسل إليه.
   ث. رسالة مصادق عليها من مكتب البريد.
   ج. رسالة مضمونة الوصول.
   ح. رسالة مسجلة بمكتب البريد.
6. حدد نوع الهجوم الأمني في كل من الحالات التالية.
   أ. طالب اقتحم مكتب الأستاذ للحصول على نسخة إمتحان اليوم الموالي.
   ب. طالب أصدر شيكا بقيمة 100 ريال لشراء كتاب، وفيما بعد اكتشف أن الشيك صرف بمبلغ 1000 ريال.
   ت. طالب أرسل مئات من البريد الإلكتروني في اليوم لطالب آخر يستعمل بريد إرجاع وهمي.
7. حدد الآلية أو الآليات الأمنية في كل من الحالات التالية.
   أ. كلية تطلب من طلابها هوية وكلمة سر للدخول على خادم الكلية.
   ب. خادم الكلية يقطع اتصال طالب يدوم أكثر من ساعتين.
   ت. أستاذ يرفض رصد درجات طالب عن طريق البريد الإلكتروني إلا إذا أظهر دليل هوية سبق للإستاذ أن أمد الطالب به.
   ث. مصرف يشترط توقيع العميل عند قيامه بعملية سحب من رصيده.

الْفَصْلُ الثَّانِي

تَقْنِيَاتُ التَّشْفِيرْ

يَهْدِفُ هَذَا الْفَصْلُ إِلَى:

1. التعريف بالمفاهيم الأساسية لعلم التشفير.
2. تقديم أهم نظم التشفير القديمة.
3. تقديم أهم نظم التشفير الحديثة.

1 مقدمة الفصل

يقول بنيامين فرانكلن: "لا يمكن أن يحتفظ ثلاثة بسر إلا إذا مات منهم اثنان". ويقول رونالد ريفست: "نتبادل الرسائل الشخصية، والمعاملات المالية، والوثائق السرية عبر الشبكة الرقمية عوضًا عن المقابلة الشخصية وجَهًا لوجه والوثائق الورقية، والأساليب التقليدية في تبادل المعلومة. كيف نتخاطب بشكل سري وخاص في حين كل المعلومات تعبر الأقمار الصناعية وتتجاوز القارات؟ كيف يمكن لبنك أن يتأكد أنه بالفعل بيل غيتس يطلب الآن من خلال محموله الخاص تحويل مبلغ 10.000.000.000 دولار لبنك آخر. لحسن الحظ تقنيات التشفير تستطيع أن تساعدنا هنا".

بدايةً وقبل الدخول في علم التشفير لا بد من تصحيح مفهوم مغلوط في أمن المعلومات، فبعض الناس يتساءل: لِمَ يحتَاج إلى تشفير بياناته في حين أنها غير ذات أهمية، ولو سئل هؤلاء عن بعض خصوصياته لامتنع عن الإجابة. فعلم التشفير هو للحفاظ على الخصوصية أيضًا. ثم إن ظن الشخص عدم امتلاكه لمعلومات مهمة غير صحيح البتة، فالمعلومات السهلة التي يستهان بها أحيانا كثيرًا ما تكون سببًا لِهَجَمَاتٍ قاتلةٍ شديدةِ الضَّرَر، وفي كل الحالات عادةً ما يكون للشخص بعض المعلومات التي يود أن تبقى سرية. وتشتد الحاجة إلى علم التشفير بديهة في التطبيقات ذات الأهمية، والحساسية العالية، كالتطبيقات العسكرية، والاقتصادية، وبعض المكتشفات العلمية وغيرها.

يمكن أن نمثل لنظام التشفير بمثال الصندوق ذي المفتاح السري. فالصندوق هو الخوارزمية، والمفتاح السري هو الذي يُمَكِّنُنَا من التعامل مع هذا الصندوق. ويقع الهجوم إما بكسر الصندوق نفسه، أو التعرف على هذا المفتاح السري. فلكي نحصل على مستوى أمنيٍّ موثوق لا بد من أن يُصْنَعَ الصندوقُ بشكل متين وقوي، ولا بد من أن يكون التعرف على المفتاح صعبًا. أي لا بد أن تكون خوارزمية التشفير قوية ومتينة، ولا بد من أن يكون حجم المفتاح طويلاً، أي عدد الاحتمالات فيه كثيرة بالقدر الكافي. وعادةً ما تتكاتف مجهودات فَرِيْقَيْ عمل في إنتاج نظام تشفير موثوق. الفريق الأول ويعبر عنه بفريق التشفير، يسعى لصناعة نظام التشفير. والفريق الثاني ويسمى فريقَ كَسْرِ نظامِ التشفير، يُرَكِّزُ مَجهودَهُ على كسر نظام

التشفير المطور. وتسمى عملية تحويل النص الأصلي لنص مشفر عملية التشفير، وعكسها عملية فَكّ التشفير.

ويعود استعمال التشفير إلى دهور بعيدة، وأول ما استعمل في التطبيقات العسكرية، فكان قواد الجيش في مدينة إسبارطة في اليونان يتبادلون المعلومات بشكل سري، وذلك باستعمال عصا تُلَفّ عليها ورقةٌ، ويَقَعُ كتابةُ الرسالة على هذه الورقة الملفوفة على العصا. ثم ترسل الرسالة التي لو تعرضت للفتح من طرف دخيل لوجد حروفها منتشرة على الورقة بشكل فوضوي، لا تلتئم إلا بِلَفّهَا مرةً ثانية على نفس العصا بنفس الطريقة. فتكون العصا هنا هي المفتاح السري الذي يجب أن يكون عند المرسل والمستقبل، وتكون الخوارزمية هي لَفّ الورقة على العصا، والكتابة على الورقة وهي ملفوفة على العصا. ثم طَوّرَ قيصرُ خوارزميته المشهورة باسمه للتواصل مع قواده العسكريين، وتعتمد على استبدال كل حرف من الأبجدية بحرف آخر، وذلك حسب قيمة سحب أو دوران ثابتة. ثم توالت عدةُ خَوارزمياتِ تَشْفيرٍ أكثر قوة، منها: خوارزمية فيجينر مع القرن السادس عشر، وتواصل استعمالها إلى نهاية القرن التاسع عشر. وتعتمد على استبدال الحرف الواحد بحروف متعددة من الأبجدية إلى أن اخترع الحاسوب، وشبكات الحاسب، وجاءت معه نُظُمُ تَشفيرٍ حديثة أكثر تعقيداً مِمّا مضى.

2 مفاهيم أساسية

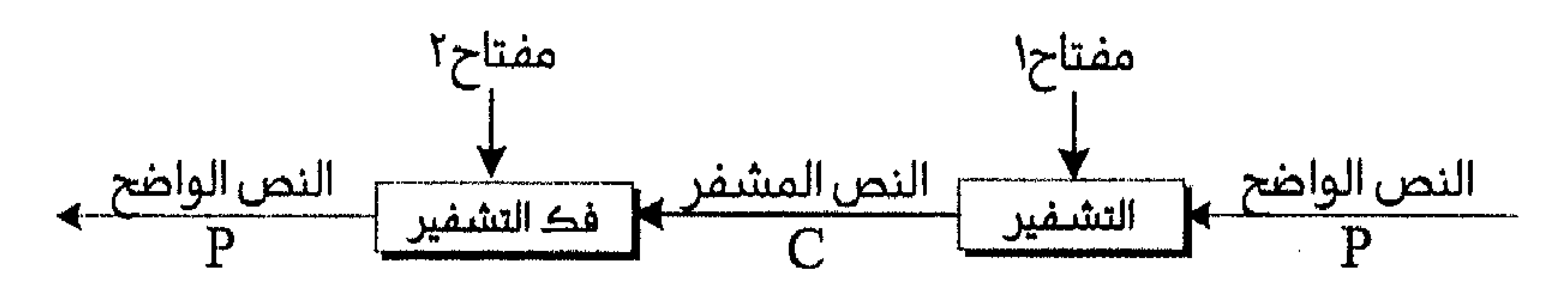
يرجع لُبّ المسألة في كيفية إجراء تبادل المعلومات بشكل آمن على قناة اتصالية، أو بيئة شبكية غير آمنة. يعتبر التبادل تبادلاً آمناً إذا كانت البيانات سريةً وسليمةً من التغيير، وهويةُ طَرَفَي التَّبادُلِ موثوقة.

تنقسم تقنية الكتابة السرية للمعلومات إلى قسمين أساسيين، وهما: التعمية والتشفير. فالتعمية تعتمد على إخفاء المعلومة دون إظهار شيء يدل على وجود سر مخفي. والتشفير يعتمد على إحداث فوضى وترميز غير مفهوم للمعلومة بحيث يصعب جداً على الدخيل فهم النص المشفر واستخراج المعلومة منه، ويعتبر التشفير التقنية الأساس في تأمين تبادل المعلومات على الشبكة اليوم. ويعتمد التشفير على عَمَلِيَتَي التبادل (transposition)، والتعويض (substitution)، وتنقسم عملية التعويض إلى نوعين أساسيين، وهما: تعويض الكلمات ويعرف بالرمز (code)، وتعويض الحروف ويعرف بالسيفر (cipher). ويعرف علم دراسة الكتابة السرية

بـ(cryptology)، وعلـم إخفـاء معلومـة فـي معلومـة أخـرى التعميـة (steganography)، وعلم الكتابة السرية بالتشفير (cryptography)، وعلم استخراج المعلومة من النص المشفر بدون معرفة مفتاح التشفير بـ(cryptoanalysis). ويستعمل مصطلح رمز (encode) أو سيفر (encipher) أو شفر (encrypt) بشكل تبادلي بالرغم من وجود اختلافات بينها في معناها الدقيق.

أمن المعلومة يرجع إلى سرية المفتاح لا إلى سرية الخوارزمية بل يقع نشر الخوارزمية للجميع. وهذا على عكس نظرية الأمن بالغموض (security by obscurity) حيث لا شيء ينشر. يقع نشر الخوارزميات للتأكد من صلاحيتها، ولمعرفة نقاط الضعف فيها، وبالتالي تحسينها. أما في الأمن بالغموض فيكمن الخطر فيما لو أن أحدًا اكتشف الخوارزمية دون أن يشعر صاحبها الذي لم يتأكد من قوتها. وهذا كمن يخفي رسالة سرية تحت مخدة نومه، وبين من يجعلها في صندوق حصين ويعرض الصندوق للناس.

شكل 2. 1 التشفير وفك التشفير



عندما يكون مفتاحا التشفير وفك التشفير متماثلين، أو يمكن استنتاج أحدهما بيسر من الآخر نتحدث على التشفير التناظري أو التماثلي، وحينما يكون مفتاحَا التشفير مُختَلِفَيْن ومن الصعب استخراج أحدهما بمعرفة الآخر، نتحدث على التشفير غير التناظري أو التشفير بالمفتاح العام وفي كلتا الحالتين لا بد أن تكون عمليتا التشفير وفَكُّه سهلة عندما تكون المفاتيح معلومة. ويمكن أن نقسم الأمن على نوعين، وهما: الأمن المطلق أو غير المشروط، والأمن المقيد؛ ففي الأمن المطلق يكون النظام آمنًا ولو فرضنا أن المهاجم يملك قدرة حسابية غير محدودة ويقاس الأمن هنا باستعمال تقنيات(information theory). أما في الأمن المشروط أو المقيد فالنظام يمكن أن يُخْتَرقَ من حيث المبدأ، ولكن هذا يحتاج أكثر من القدرة الحقيقية التي يمكن أن يمتلكها المهاجم. ويقاس الأمن هنا باستعمال تقنيات(complexity theory)

2.1 أنواع الهجوم

يمكن أن نمثل المهاجم كمنافس لنا في لعبة، يمتلك جملة من المدخلات، وهي أي شيء يمكن أن يحصل عليه المهاجم قبل بداية الهجوم، مثل: المفتاح العام، والنصوص الواضحة غير المعماة، وجملة أخرى من المعلومات التي تَصِفُ مثلاً أنواعًا من الهجومات أو نماذجها، وجملة من المخرجات، وهي أي شيء يريد المهاجم الوصول إليه كالمفتاح السري، جزء معين من النص السري ...إلخ. وعندما يتمكن من الوصول لبغيته يعتبر هذا نجاحًا له. وتنقسم أنواع الهجوم أساسًا إلى ستة أقسام مختلفة بحسب نوعية البيانات التي يمتلكها المهاجم.

1 – هجوم النص المشفر فقط:

حيث الْمُعْطَى فيه نصوص مشفرة $C_1 = E_K(M_1), \ldots, C_n = E_K(M_n)$ والهدف استنتاج $M_1, \ldots, M_n$ أو خوارزمية تحسب $M_n + 1$ من $C_n + 1 = E_K(M_n + 1)$

2 – هجوم النص الواضح المعروف:

حيث الْمُعْطَى فيه نصوصٌ مشفَّرَة، والنصوص الواضحة المقابلة لها $M_1, C_1 = E_K(M_1), \ldots, M_n, C_n = E_K(M_n)$، والهدف استنتاج مفتاح فك التشفير أو خوارزمية تحسب $M_n + 1$ من $C_n + 1 = E_K(M_n + 1)$

3 – هجوم النص الواضح المختار:

وهو نفس الهجوم السابق، إلا أن المهاجم يستطيع اختيار $M_1, \ldots, M_n$

4 – هجوم النص الواضح المختار والمتكيف:

حيث يكون المهاجم ليس قادرًا على اختيار النصوص الواضحة فقط، بل يستطيع أن يغير النص الواضح بناءًا على نتائج التشفير التي يحصل عليها.

5 – هجوم النص المشفر المختار:

حيث يمكن للمهاجم أن يختار نصوص مشفرة مختلفة لفك تشفيرها والوصول إلى النصوص الواضحة لها.

6 – هجوم التعنيف:

حيث يلجأ المهاجم للاعتداء الجسدي أو النفسي على حامل المفتاح حتى يحصل على المفتاح.

عندما نقوم بتحديد نوع الهجوم، ونعرف ما يحتاجه المهاجم للوصول لبغيته، مثل: تحديد شرط في مخرجاته التي سيحصل عليها، فهنا نصنف النظام على أنه آمن تحت هذا التعريف، إلا عندما لا يستطيع مهاجم ذو قدرات عالية النجاح في الوصول لبغيته إلا باحتمالية ضعيفة جداً. أما التعريف المعياري للأمن فيفترض عدم امتلاك المهاجم للمدخلات، ويمكن أن يقوم بهجوم على النص الواضح المختار في الحالتين التاليتين، وهما: أن يجيب النظام على هذا النص المختار بتشفيره بمفتاح معين يقع اختياره عشوائياً، أو أن يجيب النظام برسالة مختارة عشوائياً مستقلة تماماً عن النص المرسل من طرف المهاجم، والفكرة الأساس هي أنه في الحالة الثانية لا يحصل المهاجم على شيء ينفعه ألبتة، لكونه لا يستطيع أن يحدد ما إذا كانت البيانات حقيقية أو لا، وبهذا لا يمكنه المساس بالبيانات الحقيقية الفعلية.

2.2 التمثيل الرياضي

لتكن الأبجدية $\mathcal{A}$ وهي مجموعة متناهية، وليكن فضاء الرسائل أو النصوص $\mathcal{M} \subseteq \mathcal{A}^*$ ولتكن $M \in \mathcal{M}$ نصاً واضحاً، وليكن $\mathcal{C}$ فضاء النصوص المشفرة حيث إن أبجديتها يمكن أن تختلف عن $\mathcal{M}$ وليكن $\mathcal{K}$ فضاء مفاتيح التشفير.

كل $e \in \mathcal{K}$ تحدد دالة تقابل من $\mathcal{M}$ إلى $\mathcal{C}$ يرمز لها E_e والتي هي دالة التشفير، لكل $d \in \mathcal{K}$، D_d ترمز إلى دالة تقابل من $\mathcal{C}$ إلى $\mathcal{M}$ وتكون D_d هي دالة فك التشفير.

تطبيق دالة E_e هو التشفير وتطبيق دالة D_d هو فك التشفير.

كل نظام تشفير أو شفرة ينضوي على مجموعة التشفير $\{E_e: e \in \mathcal{K}\}$ وما يقابلها من مجموعة فك التشفير $\{D_d: d \in \mathcal{K}\}$ حيث يكون لكل $e \in \mathcal{K}$ مقابل وحيد $d \in \mathcal{K}$ بحيث يكون $D_d = E_e^{-1}$ أي أنه $D_d(E_e(m)) = m$ لكل $m \in \mathcal{M}$

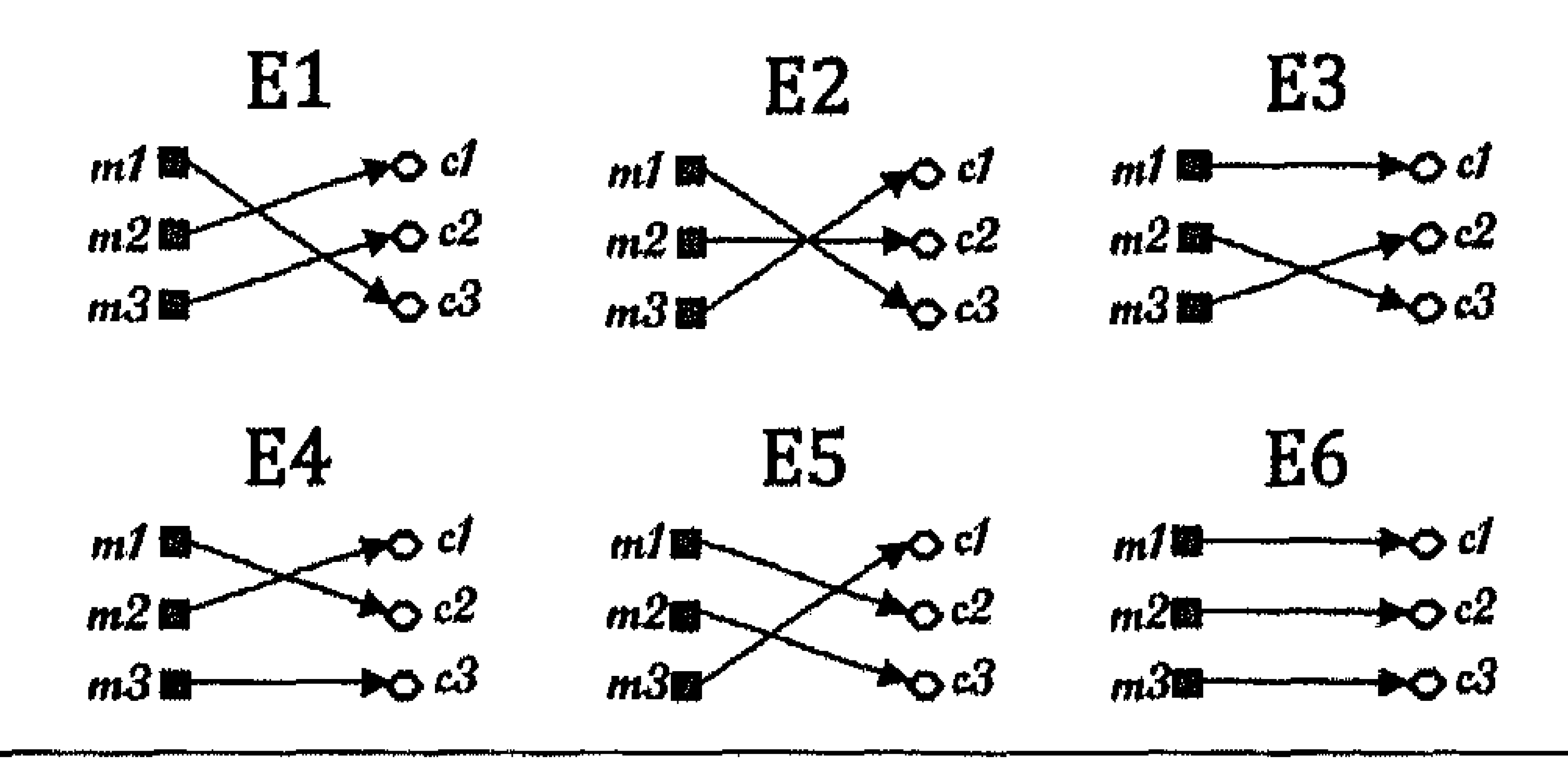
المفتاح e و d يكونان زوج المفاتيح، ويشار إليهم ب (e, d)، ويمكن أن يكونا متماثلين وعندها يكون نوع التشفير تماثلياً.

أخيرًا لصناعة شفرة نحتاج إلى تحديد $\mathcal{M}$ و $\mathcal{C}$ و $\mathcal{K}$، ومجموعة التشفير $\{E_e: e \in \mathcal{K}\}$ وما يقابلها من مجموعة فك التشفير $\{D_d: d \in \mathcal{K}\}$.

مثال توضيحي:

لتكن $\mathcal{M} =\{m_1, m_2, m_3\}$ و $\mathcal{C} =\{c_1, c_2, c_3\}$ هناك $3! = 6$ تقابل من $\mathcal{M}$ إلى $\mathcal{C}$ فيكون فضاء المفاتيح $\mathcal{K} = \{1,2,3,4,5,6\}$ يَصِفُ الشفرات الستّ.

شكل 2.2 إحتمالات التقابل بين فضاء الرسائل وفضاء النصوص المشفرة.



لنفترض أن زيدًا وعبيدًا اختارا طريقة التشفير الأولى E_1، فإن زيدًا سيقوم بحساب $E_1(m_1) = c_3$ لتشفير m_1 ويفك عبيد الشفرة c_3 بقلب السهم ليحصل على m_1 وتكون الشفرة أو نظام التشفير تماثليًّا حينما تكون المجموعتان $\{E_e: e \in K\}$ و $\{D_d: d \in K\}$ حين نستطيع لكل زوج (e, d) أن نستنتج e من d و d من e بسهولة. عمليا في التشفير التماثلي يكون $d = e$. ويطلق على التشفير التماثلي إطلاقات أخرى، منها: التشفير بالمفتاح المفرد، أو المفتاح الواحد، أو المفتاح الخاص، أو التشفير التقليدي. كما يعرف نظام تشفير الكتل (block cipher) بتقسيم النص إلى كتل ذات حجم محدد ويتم تشفير تلك الكتل كتلةً كتلةً. أما نظام تشفير الدفق (stream cipher) فيكون حجم الكتلة واحد بت فقط، أما نظام

التشفير بالترميز (codes) فيتعامل مع كلمات ذات أحجام مختلفة (انظر السطر الثاني من الشكل 2 .3).

شكل 2 .3مثال للتشفير بالترميز.

Word	Code
...	...
I	1725
discovered	5230
their	7806
are	2192
thousand	1020
...	...

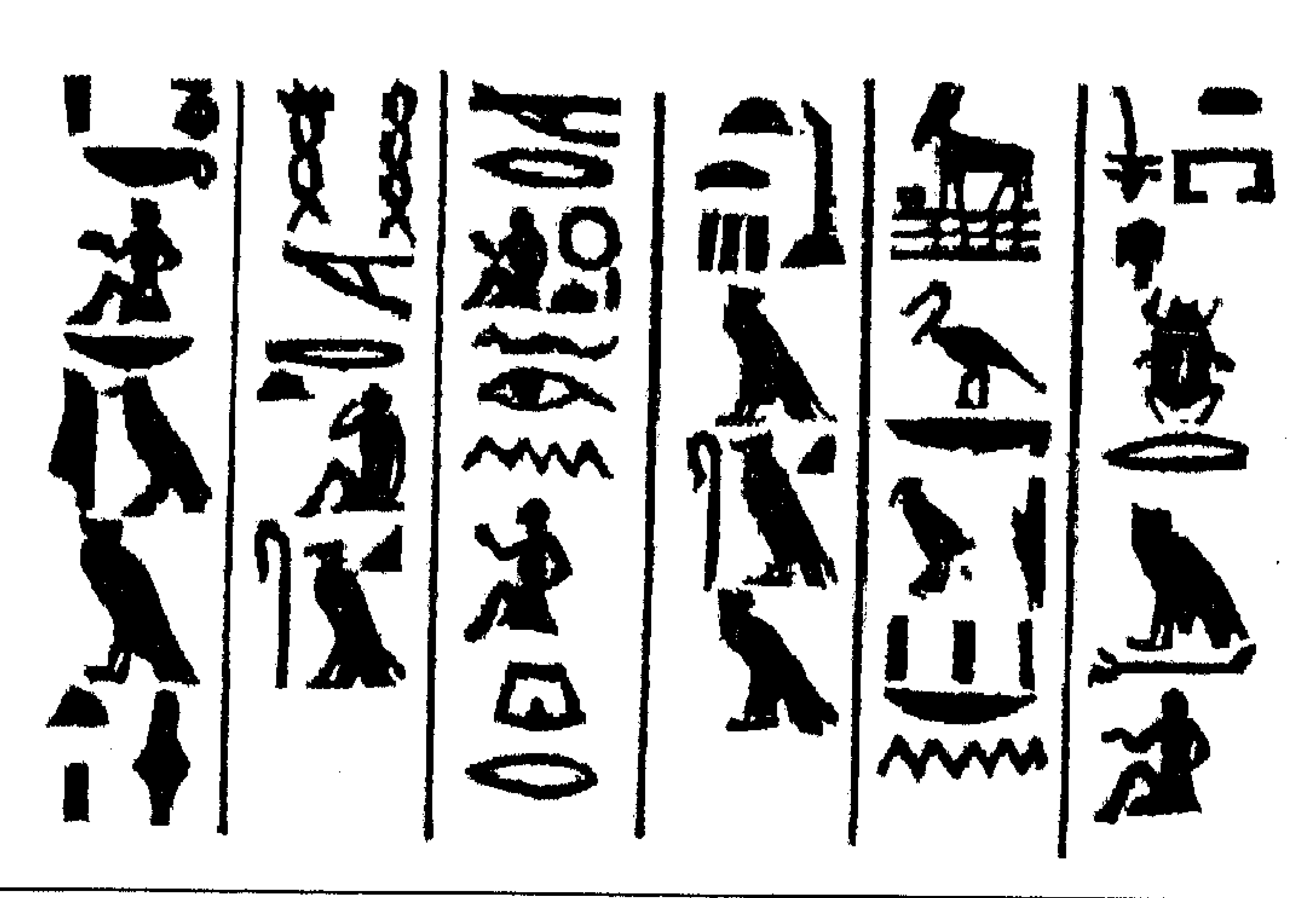
I will not say I failed thousand times =
1725 8265 0100 2001 1725 3225 1020 0001

I will say I discovered their are thousand ways that can cause failure=
1725 8265 1725 5230 7806 2192 1020 0995 6950 0385 9523 6363

3 نظم التشفير القديمة

استعمل التشفير من قبل 4000 سنة من طرف المصريين القدامى، وذلك من خلال الكتابة الهيلوغرافية (أنظر الشكل 2 .4)

شكل 2 .4 الكتابة الهيلوغرافية.



كما شفر العبريون القدامى بعض كلمات الكتب المقدسة. وقبل 2000 سنة استعمل جيليوس سيزر خوارزميته المعروفة بخوارزمية قيصر، لكن تطور التشفير كعلم مؤسس ومنظم على يد العلماء الرياضيين واللغويين العرب إبان العصر الذهبي للحضارة الإسلامية، ومن أشهرهؤلاء العلماء الفراهيدي والكندي والخوارزمي، حيث قدم هؤلاء العلماء مفاهيم

رياضية متقدمة، من أهمها التوافيق والتباديل، ساهمت في التأسيس لعلم التشفير. وكلمة التشفير وافدة من اللغات الأوربية (Cipher) و هذه بدورها جاءت أصلاً من اللغة العربية عندما اخترع العرب رقم صفر فكان هذا عند الأوروبيين اختراعًا غير مفهوم، فعبر عن كل شيء صعب الفهم بالصفر فمنه جاءت كلمة سيفر. كما ساهم العرب كما في كتاب »أدب الكتاب« في اختراع طريقة كسر الشفرة بتحليل تردد الحروف، وأول من وصفها أبو يوسف الكندي في رسالته التي عُثِرَ عليها في اسطنبول عام (1987)، والتي هي بعنوان رسالة في فك الرسائل المشفرة، وقد وصف هذه الطريقة وصفًا تَامًا. وفي عام (1200) وصف روجيه بيكون مستفيدًا من المسلمين عدة طرق للتشفير، وكذلك جوفري شوسر، ثم وضع ليون ألبرتي عجلة تشفيره الشهيرة، ووصف مبادئ التحليل بحساب تردد الحروف لكسر الشفرة في عام (1460) وسبقه إلى ذلك العرب بقرون. نستعرض في ما يلي بعض هذه الأعمال والتي يمكن تقسيمها إلى خوارزميات مبنية على عملية التعويض، وأخرى مبنية على عملية التبادل.

3.1 نظم التشفير المبنية على التعويض

3.1.1 نظام تعويض Mono-alphabetic

لتكن $\mathcal{K}$ مجموعة كل امكانيات التبديل في الأبجدية $\mathcal{A}$. نعرف لكل $e \in \mathcal{K}$ دالة التشفير E_e بحيث لكل رسالة $m = m_1m_1 \dots m_n \in \mathcal{M}$ يقابلها الرسالة المشفرة.

$$E_e(m) = e(m_1)e(m_2) \dots e(m_n) = c_1c_2 \dots c_n = c$$

وللفك شيفرة الرسالة المشفرة c يقع حساب التبديل العكسي

$$D_d(c) = d(c_1) \dots d(c_n) = m \text{ و } d = e^{-1}$$

تمثل E_e هنا نظام تشفير Mono-alphabetic.

أمثلة:

- خوارزمية قيصر:

تعوض كل حرف بالحرف الثالث الموالي له في اتجاه اليمين 26 mod في الأبجدية اللاتينية: KHOOR ZRUOG = HELLO WORLD

وبشكل عام فهي تعتمد على استبدال كل حرف من الأبجدية بحرف آخر، وذلك حسب قيمة سحب أو دوران ثابتة. فمثلا في حال أخذنا قيمة الدوران مساوية للواحد فإن كل حرف من الأبجدية سوف يقابله الحرف الذي يليه (A→B , B→C , ... , Y→Z , Z→A) وهكذا دواليك. فالمفتاح السري هو قيمة الدوران، والخوارزمية هي الاستبدال بين الحروف، وبحساب عدد الاحتمالات لقيمة الدوران وهو عدد صغير (26 احتمال للغة الانجليزية مثلا) يمكن كسر الخوارزمية واكتشاف الرسالة الأصلية بيسر.

- خوارزمية ROT13: حيث يقع سحب كل حرف بقيمة دوران 13

مثال:

Zl anzr vf Nqnz = My name is Adam

ويكفي أن تنفذ في نظام يونكس الأمر التالي:

```
$ tr a-zA-Z n-za-mN-ZA-M
```

- خوارزمية Alphanumeric: حيث يقع التعويض الأحرف برقم ترتيبها في الأبجدية.

مثال:

2 – 25 – 5 2 – 25 – 5 = BYE BYE

3.1.2 نظام تعويض Homophonic

لكل $a \in A$ نسند $H(a)$ لسلاسل الحروف المكونة من حرف t حيث تكون $H(a), a \in A$ منفصلة لكل زوجين فيها.

نظام تعويض Homophonic يعوض كل a بسلسلة حروف مختارة عشوائيًا من المجموعة $H(a)$. ولكي نفك شفرة رسالة c مكونة من t حرف، لا بد من تحديد $a \in A$ حيث تكون $c \in H(a)$ ويكون المفتاح للنص المشفر هو المجموعات $H(a)$.

مثال توضيحي:

$\boldsymbol{H(b) = \{01, 11\}}$ و $\boldsymbol{A = \{a, b\}, H(a) = \{00, 10\}}$

فالنص الواضح ab يقع تشفيره لخيار من المجموعة التالية: $0001, 0011, 1001, 1011$

3.1.3 نظام تعويض Poly-alphabetic

نظام تعويض Poly-alphabetic هو نظام تشفير بالكتل حيث حجم كل كتلة t على الأبجدية $\mathcal{A}$ بحيث يحتوي فضاء المفاتيح $\mathcal{K}$ على كل المجموعات المرتبة لـ t احتمالات تبديل على $\mathcal{A}$ وهي:$(p_1, p_2, \dots, p_t)$ فتشفير الرسالة $m = m_1 \dots m_t$ بالمفتاح $e = (p_1, p_2, \dots, p_t)$ يكون $E_e(m) = p_1(m_1) \dots p_t(m_t)$ ويكون مفتاح فك التشفير لـe هو المفتاح $d = (p_1^{-1}, \dots, p_t^{-1})$.

مثال:

- خوارزمية فيجينر (**Vigenere**): انتشرت خوارزمية فيجينر مع القرن السادس عشر وتواصل استعمالها إلى نهاية القرن التاسع عشر. وتعتمد على استبدال الحرف الواحد بحروف متعددة من الأبجدية كما وضح بالتمثيل الرياضي أعلاه.

يحسب المفتاح بسلسلة الأعداد $e = e_1, \dots, e_t$ حيث $p_i(a) = (a + e_i) \bmod n$ تعرف عملية تبديل على أبجدية بحجم n.

فمثلا بالنسبة للغة الانجليزية تكون $(n = 26)$ مع المفتاح $k = 3, 7, 10$ أي كلمة مفتاحية هي وهي DHK فلو أردنا تشفير الرسالة التالية:

m = THI SCI PHE RIS CER TAI NLY NOT SEC URE

فسنحصل على النص المشفر التالي:

$E_e(m)$ = WOS VJS SOO UPC FLB WHS QSI QVD VLM XYO

فحرف **T** يعوض بالحرف الموالي على بعد 3 فيكون **W** وحرف **H** يعوض بالحرف السابع الذي يليه وهو حرف **O** وحرف **I** يعوض بالحرف العاشر الذي يليه فهو حرف **S** وهكذا دواليك. الشكل الموالي يعرض مصفوفة ذات بعدين وهي مصفوفة فيجينير حيث أن كل سطر فيها يعتبر خوارزمية قيصر بسحب تصاعدي.

شكل 5.2 مصفوفة فيجينير.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

تُستعمل هذه المصفوفة جنبًا لجنب مع كلمة مفتاحية للتشفير لتوليد النص المشفر. فمثلا لو استعملنا كلمة **RELATIONS** ككلمة مفتاحية تكرر على طول النص الواضح، وفي كل مرة يستخرج حرف التشفير من الجدول مباشرة فتقاطع **R** مع **T** يعطينا حسب الجدول **K** وهكذا مع جميع الحروف لنحصل في النهاية على النص المشفر.

شكل 6.2 مثال التشفير بفيجينير.

	A	B	...	T	...
A	A	B	...	T	...
B	B	C	...	U	...
⋮	⋮	⋮	⋮	⋮	⋮
R	R	S	...	K	...
⋮	⋮	⋮	⋮	⋮	⋮
Z	Z	A	...	S	...

الكلمة المفتاحية: REL ATIONSRELA TIONSREL ATIONSREL ATI ONSREL AT ONS RELATI ONS REL ATION

النص الواضح: THE DIFFERENCE BETWEEN STUPIDITY AND GENIUS IS THAT GENIUS HAS ITS LIMITS

النص المشفر: KLP DBNTRJVRNE UMHJWVR DTNXWQAKC LNW OSAALW TS MPOG YVRTUL POF AKW WIFQHF

نلاحظ هنا ميزة تشفير فيجينر حيث لو نظرنا إلى أحرف **T** في النص الأصلي لوجدنا أنه تشفر بعدة حروف مختلفة، لا بحرف واحد محدد كما هو في خوارزمية قيصر.

في فك الشفرة نقوم بالعملية العكسية.

شكل 7.2 فك التشفير بفيجينير.

	A	B	...	R	...
A	A	B	...	R	...
B	B	C	...	S	...
⋮	⋮	⋮	⋮	⋮	⋮
T	T	U	...	K	...
⋮	⋮	⋮	⋮	⋮	⋮
Z	Z	A	...	Q	...

الكلمة المفتاحية: REL ATIONSRELA TIONSREL ATIONSREL ATI ONSREL AT ONS RELATI ONS REL ATION

النص المشفر: KLP DBNTRJVRNE UMHJWVR DTNXWQAKC LNW OSAALW TS MPOG YVRTUL POF AKW WIFQHF

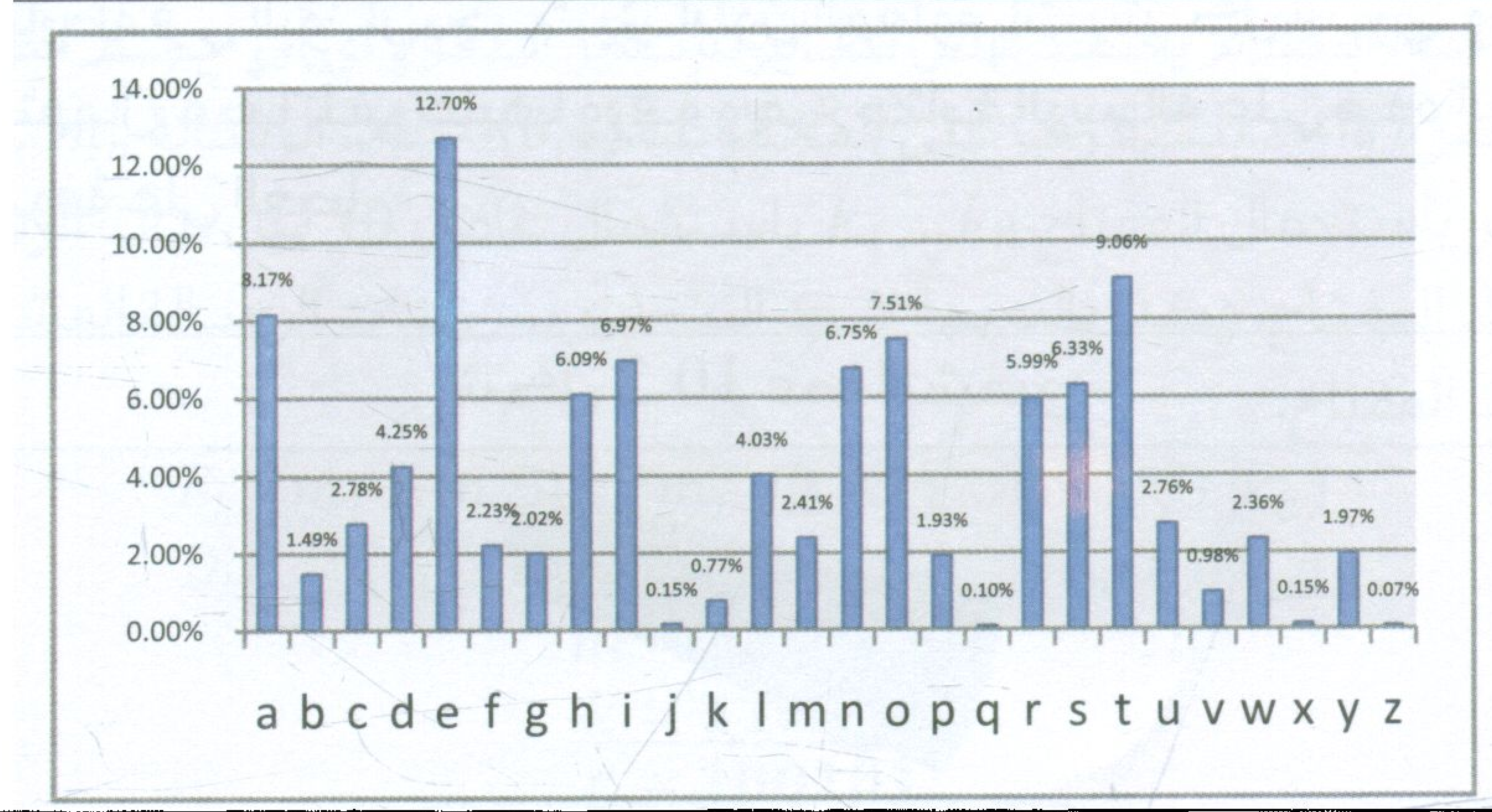
النص الواضح: THE DIFFERENCE BETWEEN STUPIDITY AND GENIUS IS THAT GENIUS HAS ITS LIMITS

تسمى الخوارزمية السابقة بخوارزمية التعويض البسيطة حيث يتم تعويض حرف وحيد بحرف آخر، بينما الخوارزمية ذات التعقيد الأكبر هي التي تقوم بتعويض مجموعات من الأحرف في الأبجدية المقروءة بعدد من الأحرف المشفرة.

تعتبر محاولة كسر خوارزمية التعويض البسيطة أكثر صعوبة من خوارزمية قيصر، ففي حين أن معرفة الحرف الأصلي المقابل لحرف مشفر في خوارزمية قيصر يؤدي لمعرفة باقي الأحرف، فإن الوضع مختلف تمامًا في خوارزمية فيجينير حيث إن مجال القيم، أو المحاولات اللازمة لكسر الخوارزمية هو: $26x25x24x \ldots x1$ وهو يساوي $26!$ ويساوي تقريبًا $4x1026$ وهو مجال كبير يوفر مناعة أكبر ضد الاختراق.

تتم محاولات الاختراق لهذه الخوارزمية بالاعتماد على تواتر الأحرف للغة الأصلية، حيث يتم حسابها عن طريق اختبار عدد كبير من النصوص، فإذا افترضنا مثلا أن تردد الحرف e هو 13% فإننا نقوم بحساب التردد للأحرف في النص المشفر فإذا وجدنا أن حرف t مثلا له تردد قريب من هذا التردد فهذا غالبا يؤدي إلى أن t في النص المشفر يقابله e في اللغة الأصلية.

شكل 8.2 تردد حروف اللغة الانجليزية.



وتفشل طرق التحليل بالتردد إذا كان النص خاصًا جدًا لا تنطبق عليه الخصائص العادية للغة، مثل أن يتكرر حرف قليل الاستعمال في نص ما يجعل تحليل التردد يؤدي إلى استنتاج خاطئ، وهو أن هذا الحرف المتكرر سيشتبه مع الحرف الأكثر تكرارًا، وهو حرف E كما هو معلوم عن اللغة الانجليزية.

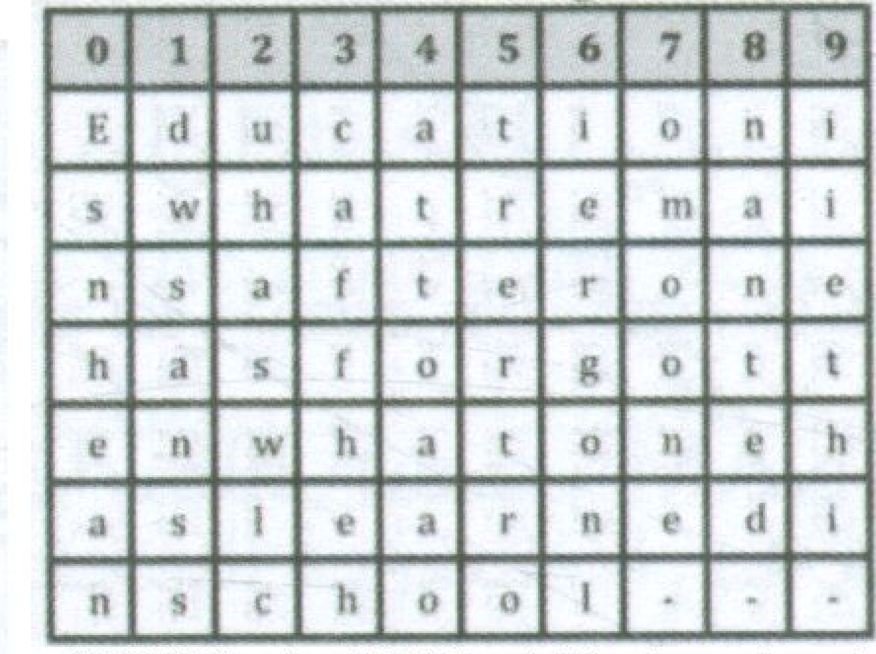
3.2 نظم التشفير المبنية على عملية التبادل

لكل كتلة بحجم t، لتكن $\mathcal{K}$ مجموعة احتمالات التبديل على $\{1, \ldots, t\}$ لكل $e \in \mathcal{K}$ و $m \in \mathcal{M}$، $E_e(m) = m_e(1)m_e(2) \ldots m_e(t)$ فمجموعة كل عملية تحويل بهذا الشكل تسمى عملية تشفير مبني على التبادل. لفك تشفير رسالة مشفرة $c = c_1c_2 \ldots c_t$ يقع حساب $D_d(c) = c_d(1)c_d(2) \ldots c_d(t)$ حيث يكون d عملية التبادل العكسية.

مثال: يعكس الشكل 9.2 عملية تبديل على 1 إلى 50 ويمثل الجدول مفتاح التشفير وفكه.

شكل 9.2 مثال لعملية تبديل على 1 إلى 50.

P = Education is what remains after one has forgotten what one has learned in school

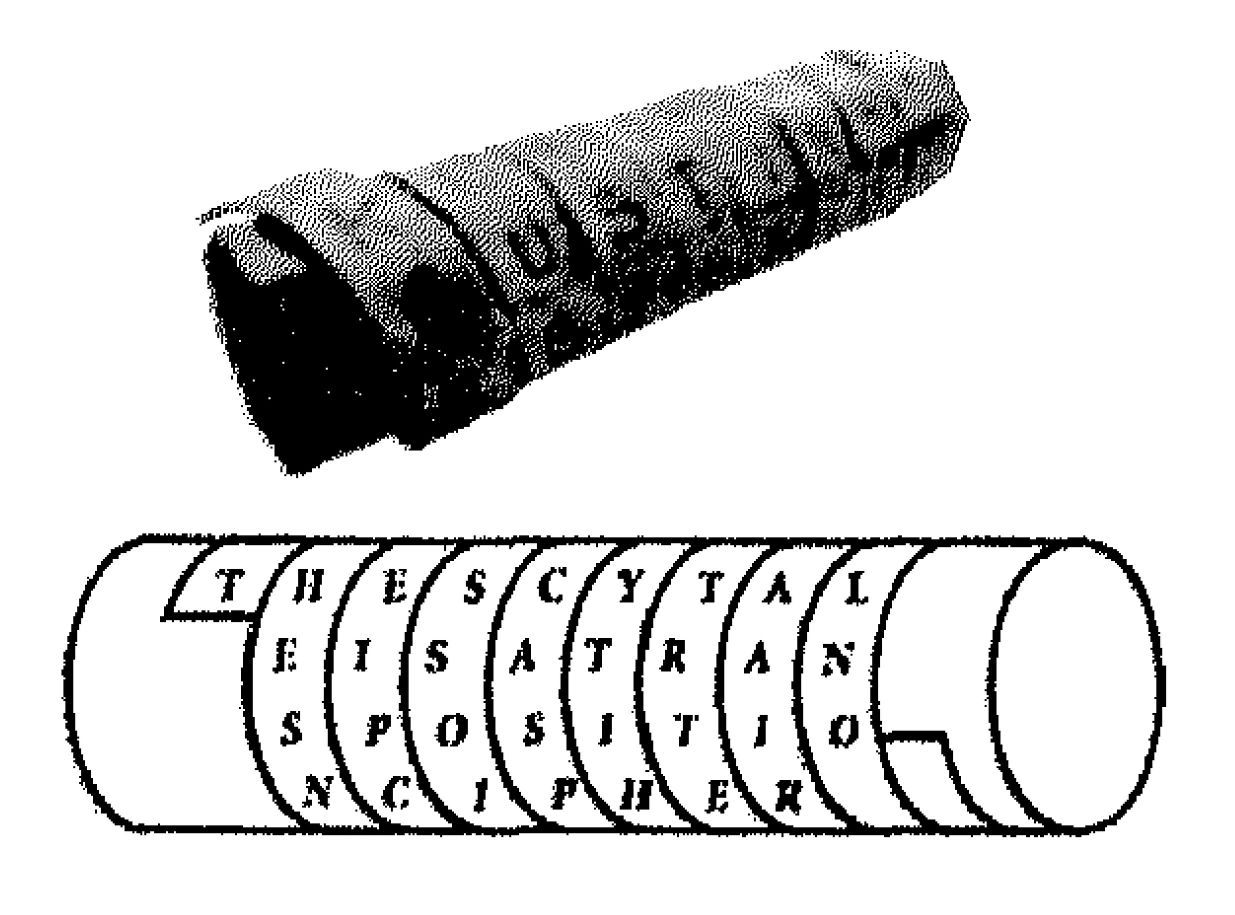


0	1	2	3	4	5	6	7	8	9
E	d	u	c	a	t	i	o	n	i
s	w	h	a	t	r	e	m	a	i
n	s	a	f	t	e	r	o	n	e
h	a	s	f	o	r	g	o	t	t
e	n	w	h	a	t	o	n	e	h
a	s	l	e	a	r	n	e	d	i
n	s	c	h	o	o	l	-	-	-

C = Esnheandwsanssuhaswlccaffhehattoaaotrertroiergonlomoonenantedilethi

وترجع جذور هذه الطريقة في التشفير إلى قُوَّادِ الجيش في مدينة إسبارطة في اليونان، حيث يتبادلون المعلومات بشكل سري، وذلك باستعمال عَصًا تُلَفُّ عليها ورقةٌ ويَقَعُ كتابةُ الرسالةِ على هذه الورقة الملفوفةِ على العصا.

شكل 10.2 عصا التشفير.



فمثلا لو جعلنا الحرف الأول مكان الحرف الثالث، والثاني مكان الأول، والثالث مكان الرابع، والرابع مكان الثاني، تشفر كلمة RENAISSANCE إلى كلمة EARNSAISCNE (انظر إلى الشكل 2. 11)

شكل 2. 11مثال للتشفير بالعصا.

$$\begin{array}{l} i \quad\;\; = 1,2,3,4 \\ f(i) = 2,4,1,3 \end{array} \Rightarrow \begin{array}{lll} \text{RENA} & \text{ISSA} & \text{NCE} \\ \text{EARN} & \text{SAIS} & \text{CNE} \end{array}$$

3.3نظم التشفير المدمجة

النظم التي تعتمد على عملية التعويض أو التبادل فقط أصبحت نُظُمًا ضعيفة اليوم. استعمال أكثر من عملية تعويض أو أكثر من عملية تبادل لا يزيد الشفرة قوة مهمة، ولكن استعمال كلتا العمليتين في التشفير يجعل النظام قويًّا بالشكل المطلوب، ومن هنا ظهر جيل حديث من نظم التشفير الذي يدمج بين العمليتين ويكررها عدة مرات بطريقة معينة، وهو ما يعرف بنظم تشفير فايستل Feistel الذي سنتعرض لشرحه في الفقرات القادمة، ولكن قبل ذلك نعرض لنظم تشفير one time pads الذي لا يمكن نظريًّا كسرها، ويعرف أيضًا بنظام تشفير VERNAM.

3.4 نظم تشفير One time pads

هـذا النظـام يعتمـد علـى نظـام تشـفير الـدفق، أي أن حجـم الكتلـة المشفرة واحد بت يعرف على المجموعـة $\mathcal{A} = \{0,1\}$، الرسـالة $m_1 \dots m_n$ تشفر باستعمال المفتاح $k_1 \dots k_n$ باسـتعمال عامـل $\oplus$ بين بتات الرسـالة وبتـات المفتـاح، وكـذا بالنسـبة لفـك التشـفير بـين بتـات الرسـالة المشـفرة وبتات المفتاح.

$$E_{k_1}(m_1 \dots m_n) = (m_1 \oplus k_1) \dots (m_n \oplus k_n)$$
$$D_{k_1 \dots k_n}(c_1 \dots c_n) = (c_1 \oplus k_1) \dots (c_n \oplus k_n)$$

$$\boldsymbol{m = 010111}$$
$$\underline{\boldsymbol{k = 110010}}$$
$$\boldsymbol{c = 100101}$$

عامل $\oplus$ ينتج للبتات المتشابهة بت0 وللمختلفة بت1.

$$\boldsymbol{0 \oplus 0 = 0}$$
$$\boldsymbol{0 \oplus 1 = 1}$$
$$\boldsymbol{a \oplus a \quad = 0\, 1 \oplus 0 = 0}$$
$$\boldsymbol{a \oplus b \oplus b = a 1 \oplus 1 = 0}$$

ونسـتعمل هـذا العامـل لأنـه ينـتج فـي احتمالاتـه الأربعـة 50% بـت 0 و505 بت 1 فلا يستطيع المهاجم التنبؤ بطبيعة النص على عكس العامل **OR** فإنـه ينـتج 75% بـت 1 و25% بـت 0 وعامـل **AND** ينـتج 75% بـت 0 و25% بت 1 فيعطي ذلك معلومة إضافية للمهاجم عن طبيعة النص.

اخترع نظام one-time pad في عام (1917) وهو نظام مثالي لا يمكن كسره نظريًا بشرط أن نستعمل في كل مرة ولمرة واحدة فقط مفتاحًا ما من سلسـلة المفاتيح المولدة عشـوائيًا، والمتبادلة بين طرفي الاتصال عن طريـق سـري موثـوق. وإلـى الآن تسـتعمل هـذا النظـام فـي الاتصـالات بـين واشنطن وموسكو، بحيث ترسـل المفـاتيح عبـر البريـد الآمـن والموثوق ثـم تسـتعمل فـي تشـفير البيانـات فـي الاتصالات اللاحقـة. تكمـن مشـكلة هـذا النظام في طرق التبادل والتزامن في اسـتعمال المفـاتيح الطويلـة بـين طرفـي الاتصال.

يمكـن أن نسـتعمل نظـام one time pad بـين الحـروف بجمـع الحـرف فـي الـنص الواضـح مـع حـرف المفتـاح الـذي يقابـل هـذا الحـرف mod حجـم الأبجدية المستعملة. فمثلا في الانجليزيـة الحجـم يكـون 26. فمثلا لـو أردنـا

تشفير هذه الرسالة ONETIMEPAD بالمفتاح التالي TBFRGFARFM فسنحصل على IPKLPSFHGQ إذ أن:

$$O + T \bmod 26 = I, N + B \bmod 26 = P$$

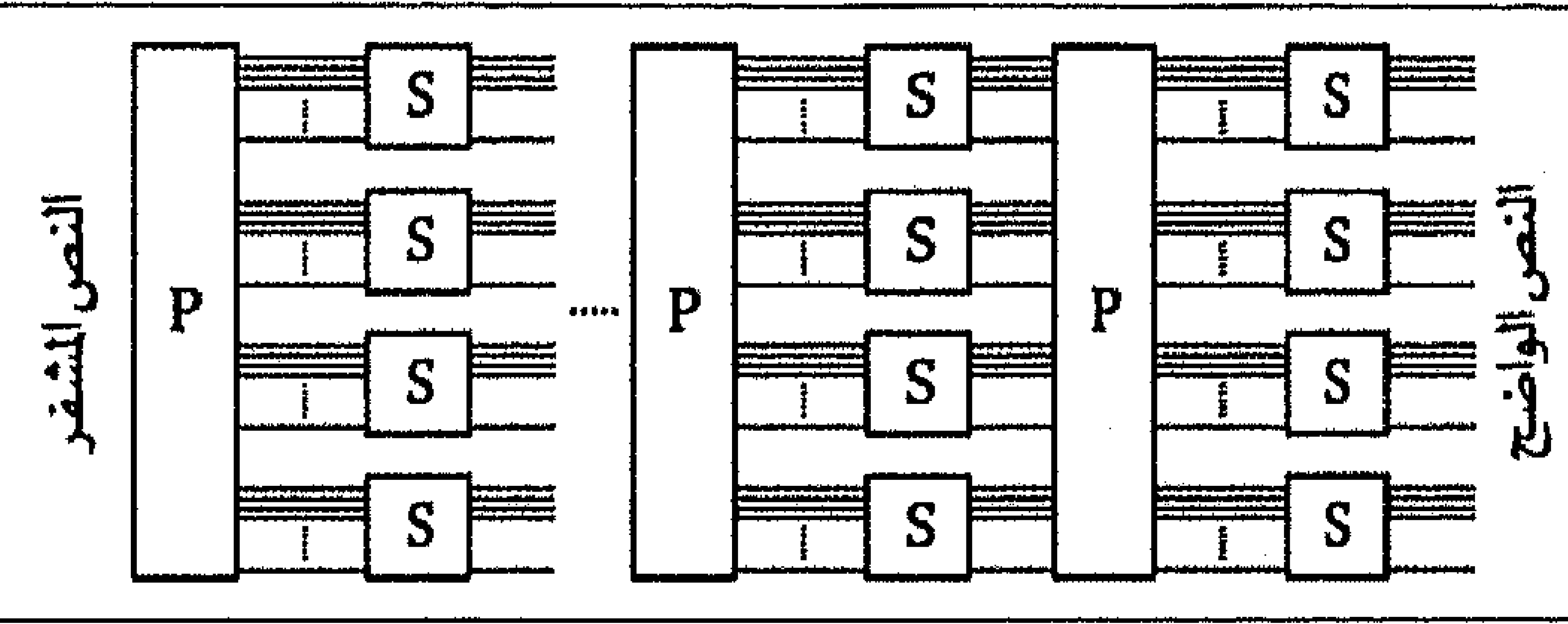
وهكذا مع بقية حروف الرسالة والمفتاح. طبعًا المفاتيح تنتج بطريقة عشوائية وتتبادل بين طرفي الاتصال بطريقة آمنة، ويكون طولها نفس طول الرسالة. ولا يستعمل إلا مرة واحدة وعندما يستعمل مفتاح ما يقع التخلص منه من كلا طرفي الاتصال، وأخيرا لا بد من التزامن بين طرفي الاتصال عند استعمال هذا النظام.

4 نظم التشفير الحديثة

4.1 التشفير التماثلي أو التناظري

نظم التشفير التي تستعمل أكثر من عملية تتكامل فيما بينها، يعبر عنها بالنظم المركبة. فمثلا سبق أن نَوَّهْنَا على أهمية استعمال عمليتي الإبدال والتعويض معًا ليكون نظام التشفير أكثر قوة. يظهر الشكل الموالي نظامًا مركبًا من عمليتي التعويض والإبدال حيث يعبر S على عملية التعويض (Substitution) التي تخلط البيانات مع بعض و P على الإبدال (Permutation) التي تفرق البيانات وتوزعها من جديد.

شكل 2. 12شبكة التعويض والإبدال.

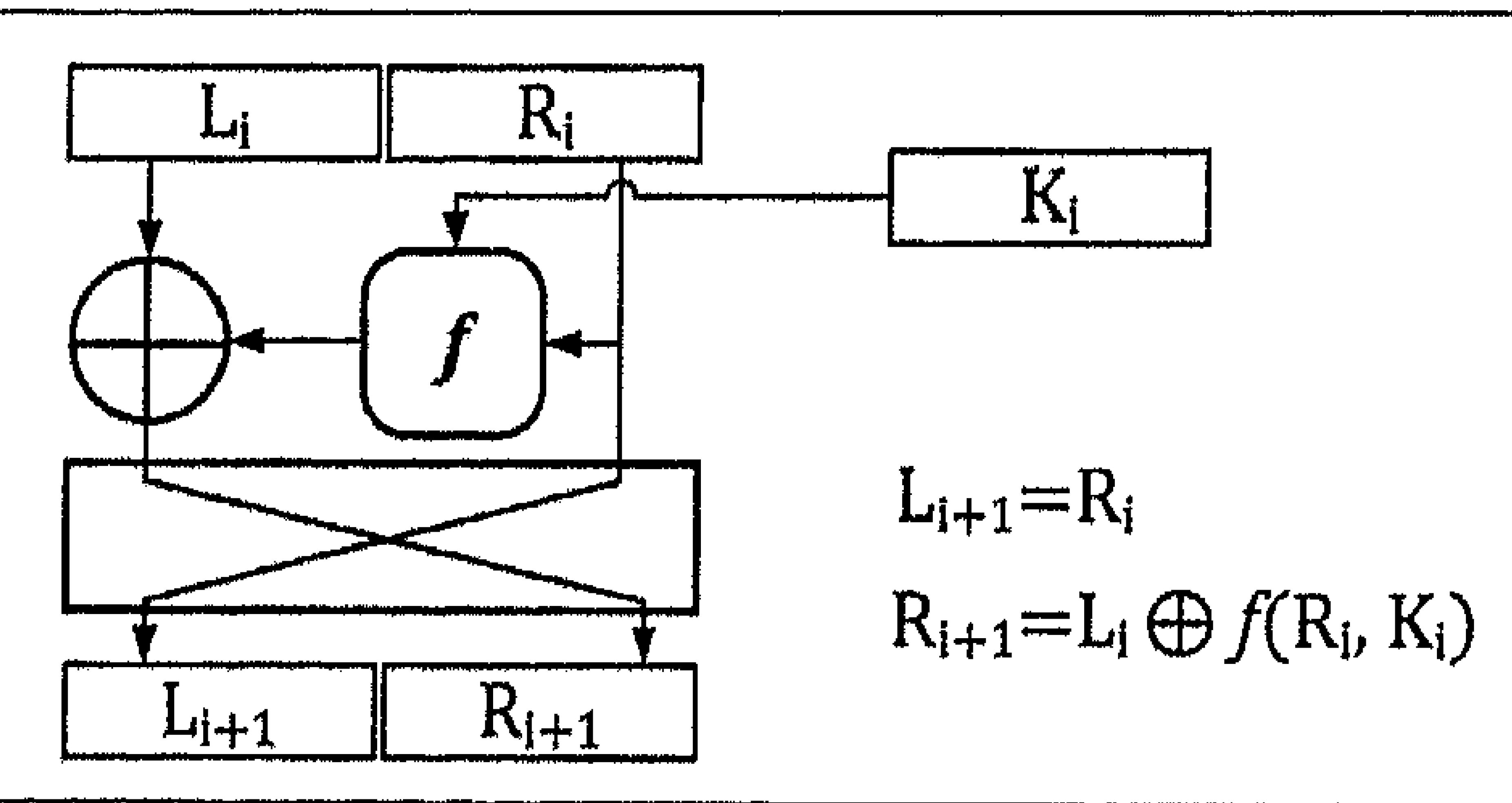


يكون نظام التشفير جيدًا وموثوقًا به إذا ما كانت هذه الشبكة من عمليتي التعويض والإبدال (S-P network) تلبي الخاصيتين التاليتين وهما:

- خاصية الكمال (Completeness) وهي: أن يكون كل بت مخرج من النظام يرتبط في توليده بكل البتات المدخلة.
- خاصية الانهيار البياني (Avalanche effect) وهي: أن وقع تغيير بت وحيد في المدخلات تغيرت المخرجات بنسبة 50% تقريبًا، ومن هنا تسمى هذه الخاصية بالانهيار الجليدي في المخرجات لتغيير طفيف جدًا في المدخلات.

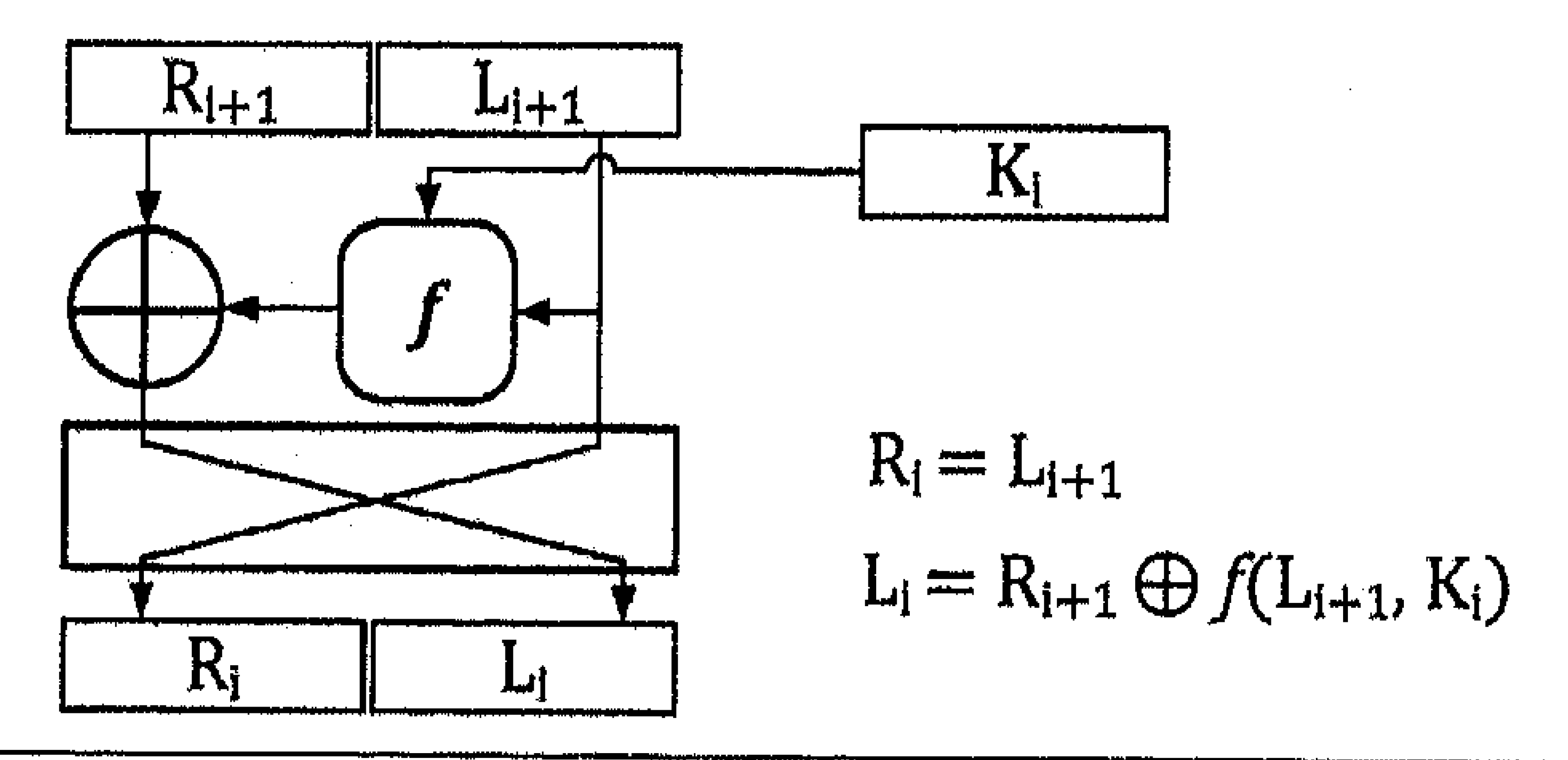
في السبعينيات أضاف هورست فيستل (Horst Feistel) من شركة IBM بُعدًا جديدًا لنظم التشفير المركبة، بجعل الشفرة تعمل على شكل جولات وتقسيم الكتل المدخلة إلى نصفين: أيمن وأيسر، وتشفير النصف الأيمن فقط بدالة التشفير التي يمكن أن تكون من نوع S-P، أو أي نوع ثان في كل جولة من جولات التشفير، ثم المبادلة بين نتيجة تشفير النصف الأيمن XOR النصف الأيسر ليصبح النصف الأيمن المدخل لجولة التشفير الموالية، وكذلك النصف الأيمن للجولة الحالية يصبح بمثابة النصف الأيسر لجولة التشفير الموالية (انظر إلى الشكل 13.2).

شكل 13.2 جولة التشفير.



وفي عملية فك التشفير نستعمل نفس المفتاح، وبنفس الجولات ولكن بالاتجاه العكسي (انظر إلى الشكل 14.2).

شكل 14.2 جولة فك التشفير.



ظهرت جملة من الخوارزميات المشهورة المبنية على فكرة فيستل يعرضها الجدول 1.2 مختلفة في حجم كتل التشفير وطول مفتاح التشفير وعدد الجولات.

جدول 2. 1 خوارزميات مبنية على عمارة فيستل

	Block size	Key Size	#Rounds
DES	64	56	16
Double-DES	64	112	32
Triple-DES	64	168	48
IDEA	64	128	8
Blowfish	64	32..448	16
RC5	32,64,128	0..2, 040	Vbl
CAST-128	64	40..128	16
RC	64	8..1, 024	16

نستعرض في ما يلي خوارزمية DES بأنواعها بالتفصيل، كنوع من أشهر أنواع نظم التشفير التناظري والتي تعتمد على بنية فيستل.

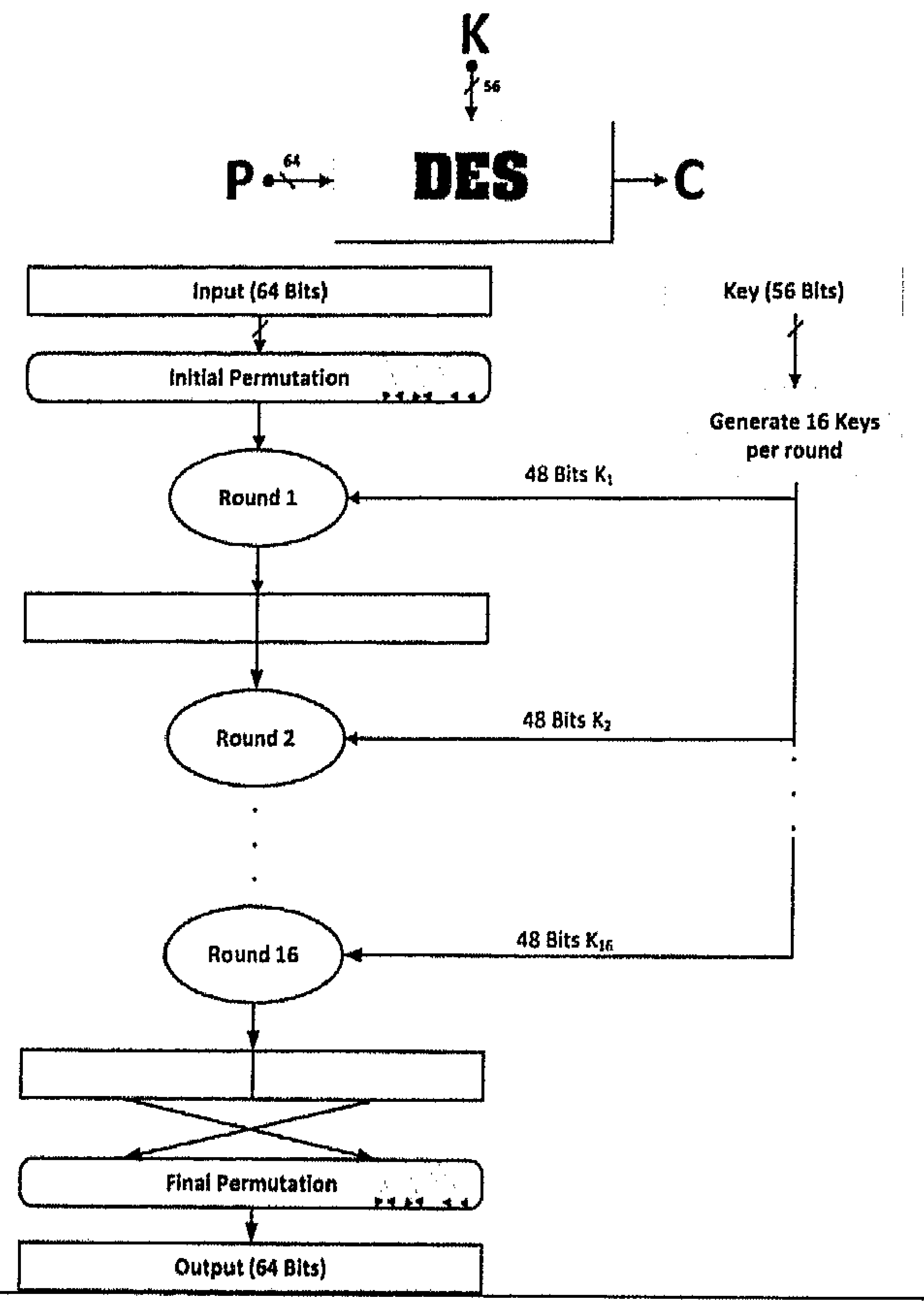
4.1.1 خوارزمية التشفير المعياري للبيانات (DES)

في 15 مايو (1973) نشرت NIST مزادًا لخوارزمية تشفير تكون مجدية وسهلة الفهم، ونشرها لا يقلل من أمنيتها، وخاصةً تضمن مستوى أمنيًا عَاليًا باستعمال مفتاح قصير الطول للتشفير وفكه. وفي عام (1974) قامت شركة IBM بتقديم خوارزمية لوسيفر Lucifer المعتمدة على نظام فيستل، الذي تم اعتماده من خلال المكتب الفدرالي الأمريكي للمعايير سنة (1977)، ثم في القطاع الخاص من منشئة ANSI سنة (1981) ليقع إقرارها لاحقًا من طرف المعهد الأمريكي للمعايير تحت اسم DES في عام 1993. استعملت هذه الخوارزمية بشكل مكثّفٍ في الأنشطة المالية خصوصا منها البنكية في أجهزة الصراف الآلي وغيرها.

1. توصيف الخوارزمية:

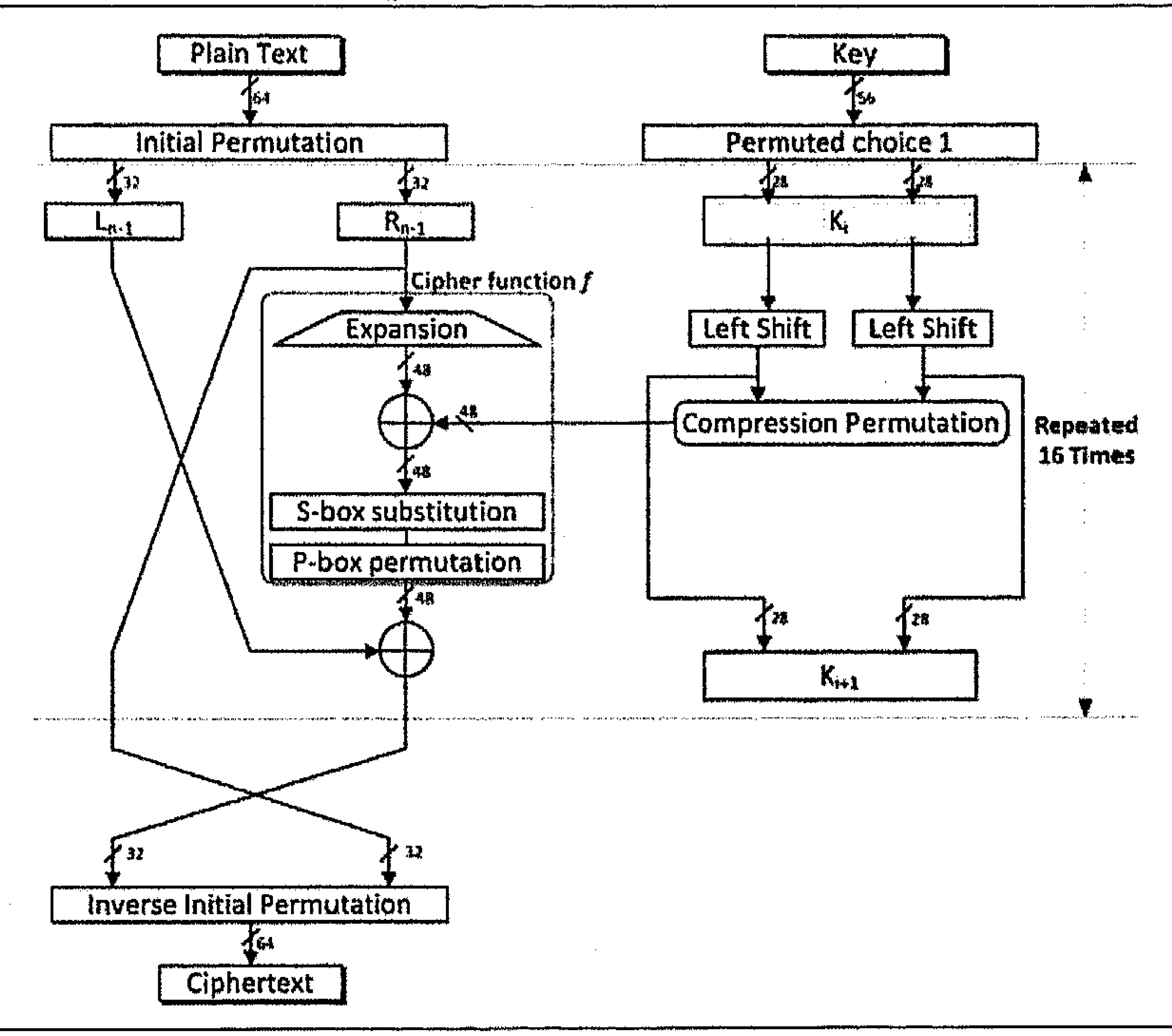
تتعامل هذه الخوارزمية مع كتل نصية بحجم 64 بت أي 8 بايت بحجم 8 بت. وتستعمل مفاتيح بحجم 8 بايت تخصص 8 بت منها لاختبار سلامة الكتلة بمقدار بت من كل بايت، فيكون المفتاح بحجم 56 بت.

شكل 15.2 عمارة خوارزمية DES.



تعتمد الخوارزمية على عمليات التعويض والتبديل التي تعمل في اتجاهي التشفير وفكه، فنقوم بعملية تبديل أولية لبتات الكتلة ثم تقسم كتلة 64 بت إلى شقين أيمن R، وأيسر L، بحجم 32 بت لنقوم بعمليتي تبديل وعملية تعويض واحدة في دالة التشفير f في كل جولة. ثم المبادلة بين نتيجة تشفير النصف الأيمن XOR النصف الأيسر، ليصبح النصف الأيمن المدخل لجولة التشفير الموالية، وكذلك النصف الأيمن للجولة الحالية يصبح بمثابة النصف الأيسر لجولة التشفير الموالية. نكرر نفس هذه العمليات في 16 جولة ليتم أخيرا تأليف الشقين الأيمن والأيسر، ثم القيام بعملية تبديل نهائية التي هي عبارة عن عملية التبديل الأولية ولكن عكسية. بالتوازي يتم توليد مفتاح التشفير بحجم 48 بت لكل جولة اعتمادًا على طريقة في التعامل مع المفتاح الأول الذي هو بحجم 56 بت (انظر إلى الشكل 16.2).

شكل 16.2 جولة التشفير في DES.



نعرض في الفقرات التالية للتوصيف المفصل للخوارزمية مع ضرب مثال تطبيقي لها.

2. توليد المفاتيح:

بعد نزع 8 بت وهي البت الأخير من كل بايت نقوم بعملية إبدال حسب المصفوفة التالية، وهي تعرف بعملية إبدال الخيار الأول (Permuted Choice 1:PC-1) حيث يكون البت رقم 57 في البت رقم 1 والبت رقم 49 في البت رقم 2 وهكذا إلى البت رقم 4 في البت رقم 56 (انظر إلى الجدول 2.2)

جدول 2.2 مصفوفة عملية إبدال الخيار الأول.

57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

ثم تقع قسمة المفتاح إلى $C[0]$صفين و $D[0]$ طول كل واحد منهما 28 بت

لتوليد 16 مفتاح للجولات الستة عشر يقع في كل جولة سحب المفتاح الأصلي إلى اليسار إما ب2 بت أو ب 1 بت حسب رقم الجولة وذلك حسب الجدول 2. 3.

جدول 2. 3 قدر سحب المفتاح.

Iteration #	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Left Shifts	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

فمثلا في الجولة الأولى يقع السحب إلى اليسار ب 1 بت، وفي الجولة الثالثة ب 2 بت.

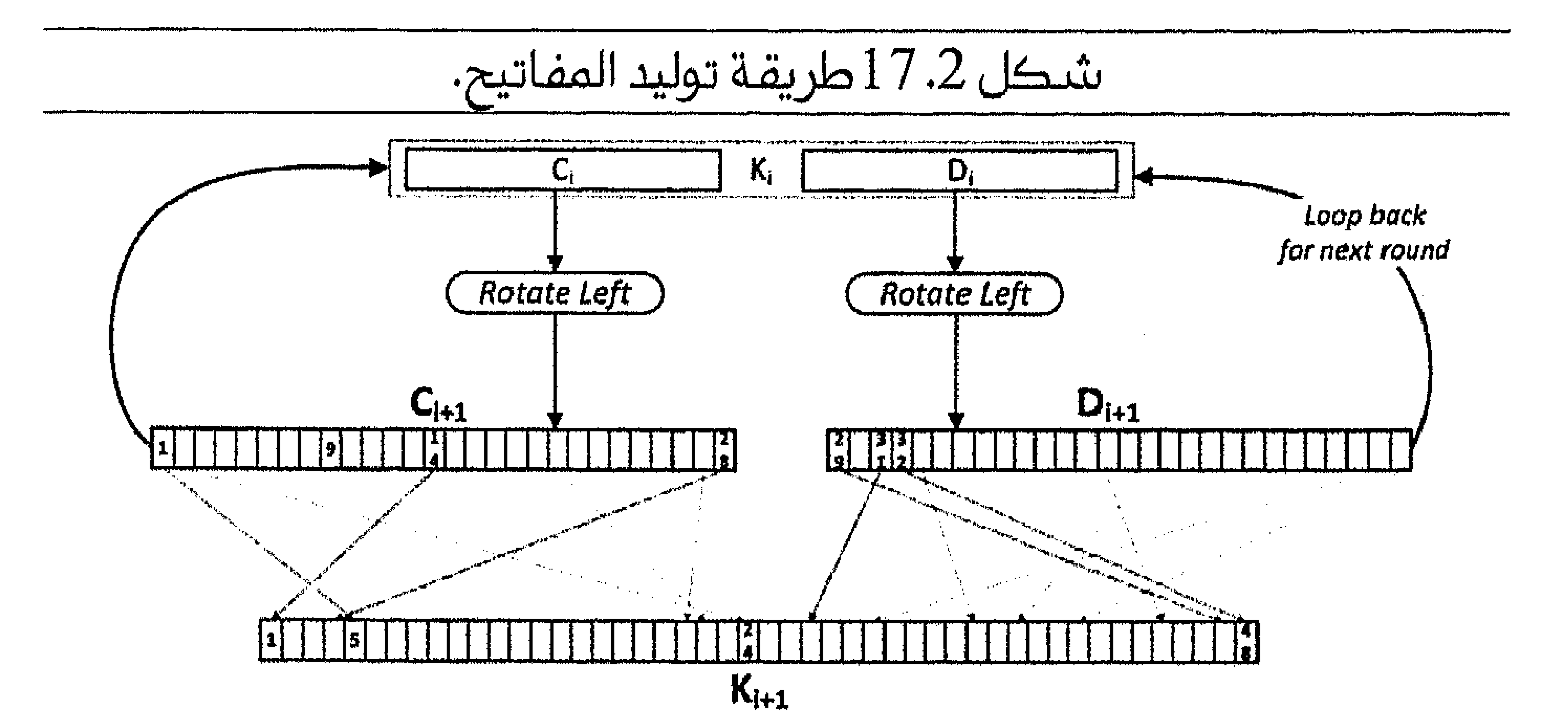
نولد المفتاح الفعلي الذي سيقع به التشفير لكل جولة بنزع 8 بت من أصل 56 بت، لنحصل على مفتاح بطول 48 بت. فننزع من النصف الأيمن من المفتاح البت رقم 38،35، 54،43، ومن النصف الأيسر البت رقم 22،18،9، 25ثم نخضع الباقي لعملية إبدال ثانية تعرف بـ(Permuted Choice 2:PC-2) (انظر إلى الجدول 2. 4)

جدول 2. 4 مصفوفة عملية إبدال الخيار الثاني.

14	17	11	24	1	5
3	28	15	6	21	10
23	19	12	4	26	8
16	7	27	20	13	2
41	52	31	37	47	55
30	40	51	45	33	48
44	49	39	56	34	53
46	42	50	36	29	32

يلخص الشكل 2. 17 طريقة توليد المفاتيح المفصلة بالخطوات أعلاه لكل جولة.

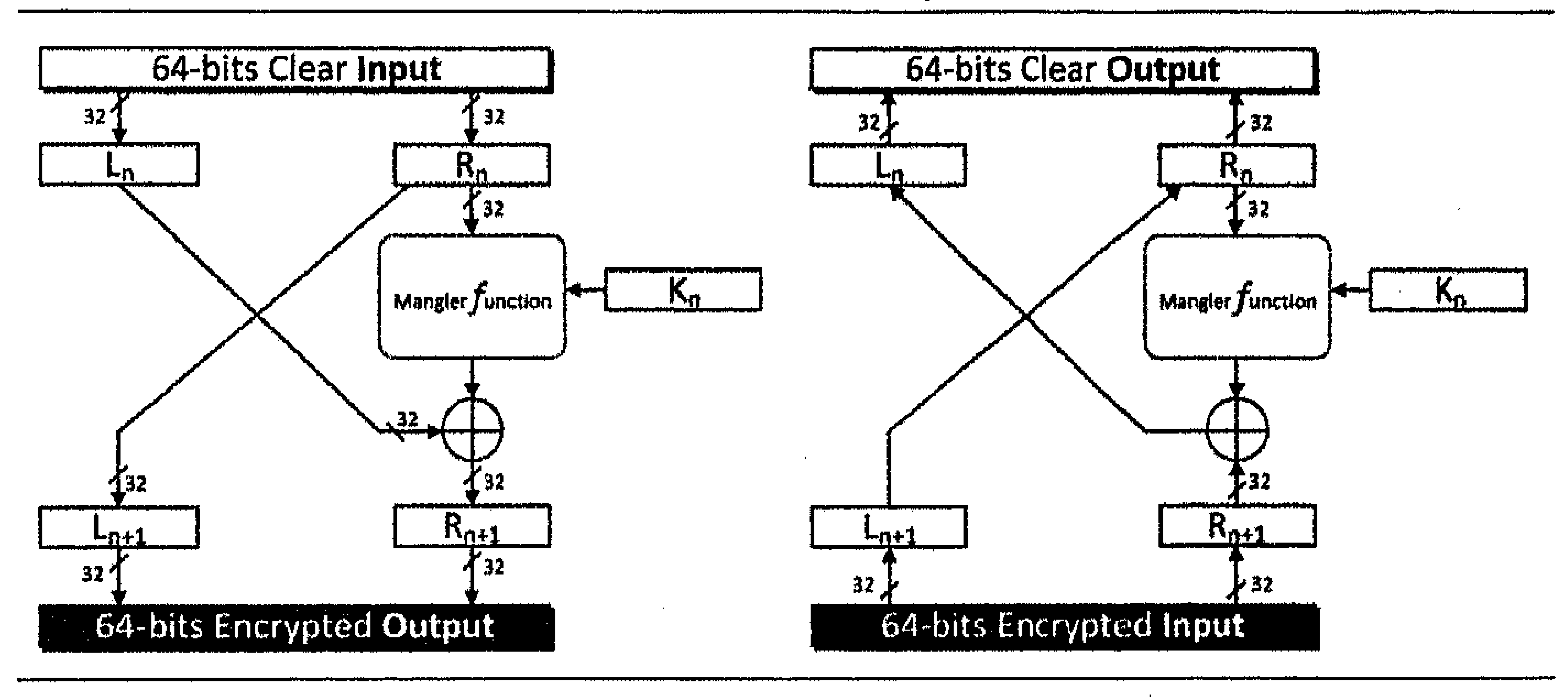
شكل 2.17 طريقة توليد المفاتيح.



3. جولة تشفير DES:

يلخص الشكل 2.18 العمليات الموجودة في كل جولة من جولات DES الستة عشر عند التشفير (الجزء الأيسر من الشكل 2.18)، وفي فك التشفير (الجزء الأيمن من الشكل 2.18) وسنعرض لهذه العمليات بالتفصيل في هذه الفقرة.

شكل 2.18 جولات DES عند التشفير و فكه.



تتم عملية التبديل الأولية باستعمال المصفوفة التالية (انظر إلى الجدول 2.5) حيث تأخذ البت رقم 58 في الكتلة المعدة للتشفير، وهي بحجم 64 بت مكان البت رقم 1، وكذا البت 50 مكان البت 2، وهكذا على البت رقم 7 مكان البت رقم 64.

جدول 2. 5 مصفوفة عملية التبديل الأولية.

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

يقع تقسم الكتلة إلى نصفين أيمن$R[0]$ وأيسر $L[0]$حجم كل واحد منهما 32 بت ثم عقب كل جولة نحصل على شقين أيمن وأيسر بنفس الحجم.

يقع توسيع حجم النصف الأيمن من 32 بت إلى 48 بت ليتلاءم مع حجم مفتاح التشفير الذي هو بحجم 48 بت. عمليًا يقع تقطيع النصف الأيمن إلى 8 كتل بحجم 4 بت الكتلة الواحدة فتكون الكتلة الأولى مركبة من البتات 1–2–3–4، والثانية 5–6–7–8 وهكذا للكتلة الثامنة وهي 29–30–31–32. ثم يقع زيادة 2بت واحد عن يمين الكتلة والثاني عن يسارها لتصبح حجم الكتلة 6. وبما أن لدينا 8 كتل فيصبح حجم النصف الأيمن 8*6=48 بت. البت المضاف عن يمين كل كتلة هو البت الأول من الكتلة الوالية لها من جهة اليمين، والبت المضاف عن يسار الكتلة هو البت الأخير من الكتلة السابقة لها. فالكتلة الثانية مثلاً وهي 5–6–7–8 تصبح 4–5–6–7–8–9 وهكذا للكتل التي في الوسط أما الكتلتان الطرفيتان فتكون البت الأولى على يمين آخر كتلة، والبت 32 على يسار أول كتلة. لنحصل في النهاية على المصفوفة التالية:

جدول 2. 6 الكتلة النصية بعد عملية التوسعة.

32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

نقوم بعملية XOR بين المفتاح وبين النصف الأيمن الذي وقع توسيعه، ونقسم الناتج إلى 8 كتل، حجم كل كتلة 6 بت.

نقلص حجم كل كتلة من هذه الكتل الثمانية من 6بت إلى 4 بت باستعمال مصفوفات التعويض الثمانية التالية (انظر إلى الشكل الموالي). نحاذي بين البت الأول والأخير من الكتلة للحصول على رقم السطر في المصفوفة، والبتات الأربعة الوسطى تحدد رقم العمود، ثم نستخرج القيمة التعويضية الموجودة في المصفوفة عند تقاطع السطر والعمود المحددين. حجم هذه القيمة التعويضية4بت، وهكذا نكون قد قلصنا الكتلة من 6 إلى 4بت. فمثلاً لو كانت الكتلة الأولى هي 001010 فإن رقم السطر سيكون بالتمثلي ثنائيّاً00 أي رقم 0 بالتمثيل العشري ورقم العمود 0101 أي العمود رقم 5 بالتمثيل العشري فنحصل باستعمال المصفوفة التعويضية الأولى على القيمة التعويضية 15 بالتمثيل العشري، أي 1111 بالتمثيل الثنائي (ملاحظة: ترقيم السطور والأعمدة يبدأ من 0). تستعمل المصفوفة التعويضية الأولى مع الكتلة الأولى، والمصفوفة التعويضية الثانية مع الكتلة الثانية، وهكذا إلى الثامنة مع الكتلة الثامنة.

جدول 7.2 مصفوفات التعويض.

Substitution Box 1 (S[1])

14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
4	1	4	8	13	6	2	11	15	12	9	7	3	10	5	0
15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

S[2]

15	1	8	14	6	11	3	4		7		13	12		5	0
3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9

S[3]

10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8
13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1
13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7
1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12

S[4]

7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15
13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9
10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4
3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14

S[5]

2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6
4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3

S[6]

12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11
10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8
9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6
4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13

S[7]

4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1
13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6
1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2
6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12

S[8]

13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7
1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8
2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11

يقع تنفيذ عملية إبدال على الناتج بعد التقليص باستعمال المصفوفة التالية (انظر إلى الجدول 2. 8)

جدول 2. 8مصفوفة الإبدال.

16	7	20	21
29	12	28	17
1	15	23	26
5	18	31	10
2	8	24	14
32	27	3	9
19	13	30	6
22	11	4	25

نقوم بعملية XOR بين الناتج من عملية الإبدال السابقة مع النصف الأيسر من البيانات لنحصل على النصف الأيمن للجولة الموالية، كما يأخذ النصف الأيسر للجولة الموالية قيمة النصف الأيمن قبل دخوله جولة التشفير الحالية.

يقع تكرار كل هذه العمليات في 16 جولة.

تقع المبادلة بين النصفين الأيمن والأيسر، ثم عملية الإبدال العكسية لعملية الإبدال الأولية حسب المصفوفة التالية:

جدول 9.2 مصفوفة عملية الإبدال الأخيرة.

40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

هكذا تتم عملية تشفير كتلة بحجم 64 بت لفك التشفير نقوم بنفس العمليات، ولكن باستعمال المفاتيح بالترتيب العكسي. فنستعمل المفتاح 16 للجولة الأولى، و15 للثانية، إلى المفتاح 1 للجولة 16. يعرض الشكل الموالي تلخيصًا برمجيًا لهذه العمليات لمرحلَتي التشفير وفكه في خوارزمية DES. ويوجد في الملحق 2 مثال تطبيقي عملي على هذه الخوارزمية.

شكل 19.2 خوارزمية DES.

IP: *Initial Permutation*
PC1: *Permuted Choice 1*
LS: *Left Shift*
FP: *Final Permutation*
PC2: *Permuted Choice 2*

Key Generation:
C[0]D[0] = ***PC1***(key)
For i from 1 **to** 16 **Do**
C[i]=***LS***[i](C[i-1])
D[i]=***LS***[i](D[i-1])
K[i] = ***PC2***(C[i]D[i])
End For

Encipherment:
L[0]R[0] = ***IP***(plain block)
For i from 1 **to** 16 **Do**
L[i]=R[i-1]
R[i]=L[i-1] XOR *f*(R[i-1], K[i])
End For
cipher block = ***FP***(R[16]L[16])

Decipherment:
R[16]L[16] = ***IP***(cipher block)
For i from 16 **to** 1 **Do**
R[i-1]=L[i]
L[i-1]=R[i] XOR *f*(L[i], K[i])
End For
Plain block = ***FP***(L[0]R[0])

4.1.2 كسر شفرة DES ونظام 3DES

في عام (1977) اقترح Diffie و Helmann صناعة آلة خاصة لكسر DES قدّرا تكلفتها بـ20 مليون دولارًا حينها. في عام (1993) اقترح MikeWiener طريقة لكسر المفاتيح الست عشر بالتوازي، أي في ذات الوقت. في 29 يناير (1997) أطلق مختبر RSA تحديًا لكسر شفرة DES لرسالة تحوي على ثلاث كتلات وهي: "The unknown message is" شارك فيها 70000 نظام ليقع فك الشفرة بعد 96 يوم من بداية التحدي. وفي يوليو (1998) فازت مؤسسة EEF بجائزة مختبر RSA بكسرها الشفرة في 56 ساعة من خلال آلة خاصة تتمكن في المتوسط باكتشاف مفتاح DES في أقل من خمسة أيام بكلفة 250000 دولار. وفي عام (1999) كُسِرَ في 22 ساعة وربع الساعة، ثم في عام 2006 كسر DES بآلة كلفتها 1000 دولار في 10 أيام لتطور بعد عام لتصل إلى 6.4 يوم، ثم طورت آلة أخرى بكلفة أقل في عام 2008 تكسر DES في أقل من يوم. على كل حال ومنذ خروج DES للوجود كان هناك شكوك قوية وأسئلة تحوم حول سبب نزع الثمانية بت من المفتاح، مع أنها تقلل من قوة الخوارزمية بإنقاص عدد الاحتمالات الممكنة، وكذلك حول المصفوفات التعويضية وطريقة تحديدها، إذ يعتقد أن الحكومة الأمريكية وضعت من خلالها أبوابًا خلفيةً يسهل عليها كسر شفرة DES لأي رسالة. سعت المؤسسات بالاستعاضة عن نظام DES بتوليد نظامين يعتمدان على DES وهما 2DES و3DES. وفي كلا الخوارزميتين نستعمل DES أكثر من مرة، ففي 2DES نستعمل مرتين DES في التشفير بمفتاحين مختلفين.

$$x \rightarrow E_{K_1}(x) \rightarrow E_{K_2}(E_{K_1}(x))$$

وفي 3DES نستعمل ثلاث مرات DES بثلاثة مفاتيح مختلفة، أو بمفتاحين مختلفين مع جعل مرحلة فك تشفير، باستعمال المفتاح الثاني بين مرحلتي تشفير بالمفتاح الأول.

$$x \rightarrow E_{K_1}(x) \rightarrow E_{K_2}\left(E_{K_1}(x)\right) \rightarrow E_{K_3}\left(E_{K_2}\left(E_{K_1}(x)\right)\right)$$

$$x \rightarrow E_{K_1}(x) \rightarrow D_{K_2}\left(E_{K_1}(x)\right) \rightarrow E_{K_1}\left(D_{K_2}\left(E_{K_1}(x)\right)\right)$$

فائدة وضع عملية فك تشفير في 3DES لأغراض التوافقية مع DES، إذ يمكن أن نولد رسالة مشفرة بـDES بجعل المفتاح الأول مساويًا للمفتاح

الثاني. باستعمال مفتاحين نكون قد جعلنا DES أكثر أمانًا إذ أصبح طول مفتاح التشفير 112.

4.1.3 تشفير النصوص الطويلة بـ DES أو 3DES

سلف القول بأن DES يعمل على تشفير كتلة نصية واحدة بحجم **64** بت، ولكن كيف لنا أن نشفر نصًا طويلاً يتكون أكثر من كتلة نصية. ظهرت عدة من الطرق سنعرض هنا لاثنين منها فقط، وهما طريقة الترميز الإلكتروني، وطريقة الربط بالكتل المشفرة.

1. طريقة الترميز الإلكتروني: وهي الطريقة البديهية، ولها عدة مساوئ. تقوم هذه الطريقة بتقسيم الرسالة إلى كُتَلٍ نصية تعادل 64 بت، ومن ثم تشفر كل كتلة نصية باستخدام المفتاح السري. الطرف الآخر يستقبل الكتل المشفرة، ويفك تشفير كل كتلة نصية من أجل الحصول على الرسالة الأصلية. هذه الطريقة فيها عيبان خطيران.

أولاً: في حال نظر شخص ما للشفرة النصية بإمكانه الحصول على معلومات من الكتل المكررة، إنه في حال كانت الرسالة تحتوي على كتل متطابقة مكونة من 64 بت عندئذ تكون شفرة هذه الكتل متطابقة.

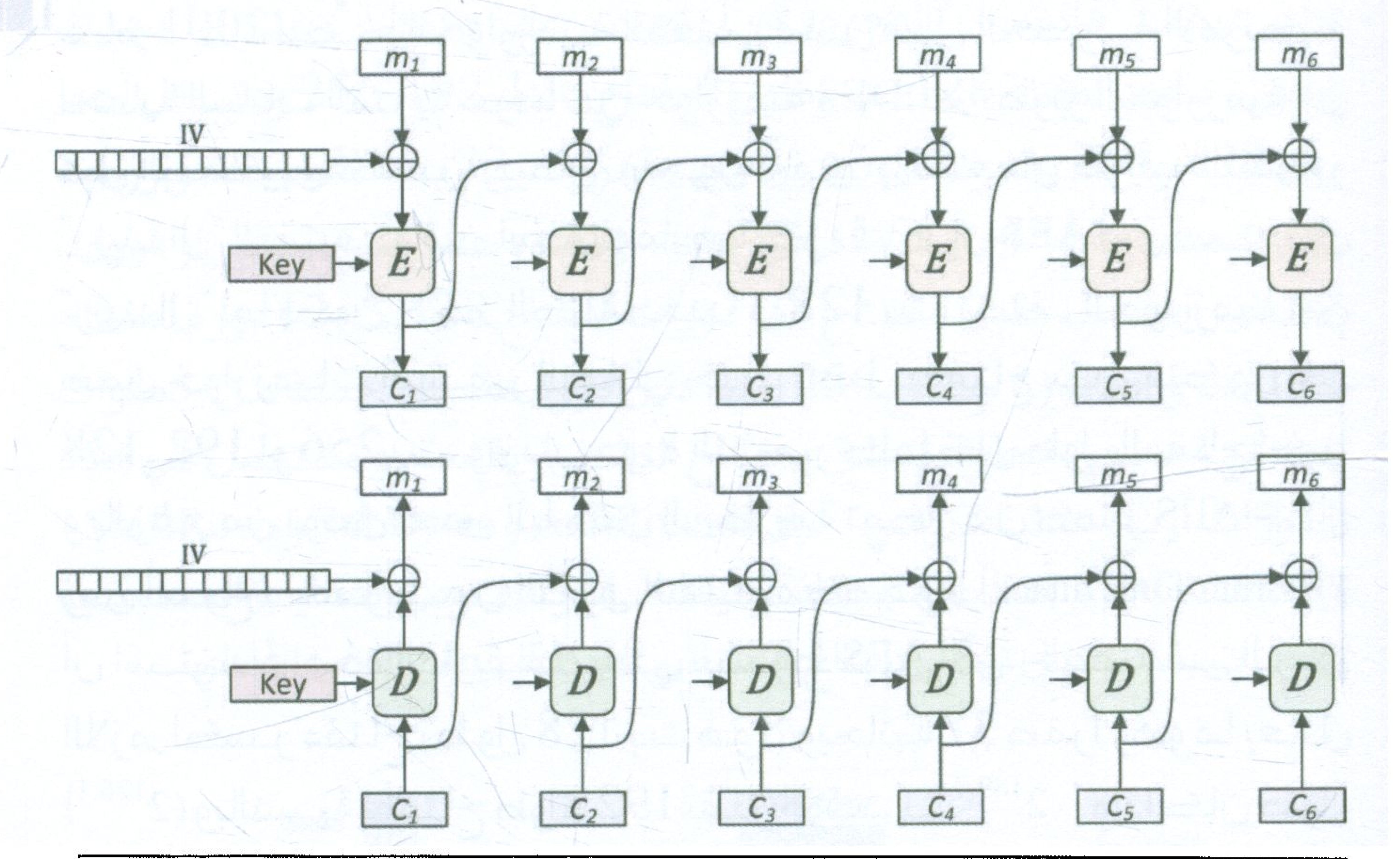
ثانيًا: يستطيع هذا الشخص أن يعيد ترتيب تلك الكتل، أو أن يعدل كتلاً لمصلحته إن كان يعرف هيكلة النص الأصلي. كذلك عملية فك الشفرة لا تعكس أي معلومة عن سلامة الكتل المشفرة من المسح والتغيير والتكرار.

2. طريقة الربط بالكتل المشفرة (CBC): تقسم هذه الطريقة النص لكتل بحجم 64 بت، ثم تربط الكتلة المشفرة السابقة مع الكتلة النصية الموالية بعامل XOR ليتم تشفير الناتج الذي يتم ربطه بنفس الطريقة مع الكتلة النصية الموالية، وهكذا حتى نهاية الرسالة. طبعًا بداية لتوليد أول كتلة مشفرة نستعمل قيمة أولية IV مع أول كتلة نصية، وهذه القيمة لا بد أن يعرفها المستقبل لكي يستعملها في فك الشفرة لاحقًا (انظر إلى الشكل 2. 20 للتشفير وفكه عن طريق هذه الطريقة).

شكل 2.20 طريقة الربط بالكتل المشفرة(CBC)

Encryption: $C_i = E_K(P_i \oplus C_{i-1}), C_0 = IV$ (Initialization Vector)

Decryption: $P_i = D_K(C_i) \oplus C_{i-1}, C_0 = IV$ (Initialization Vector)



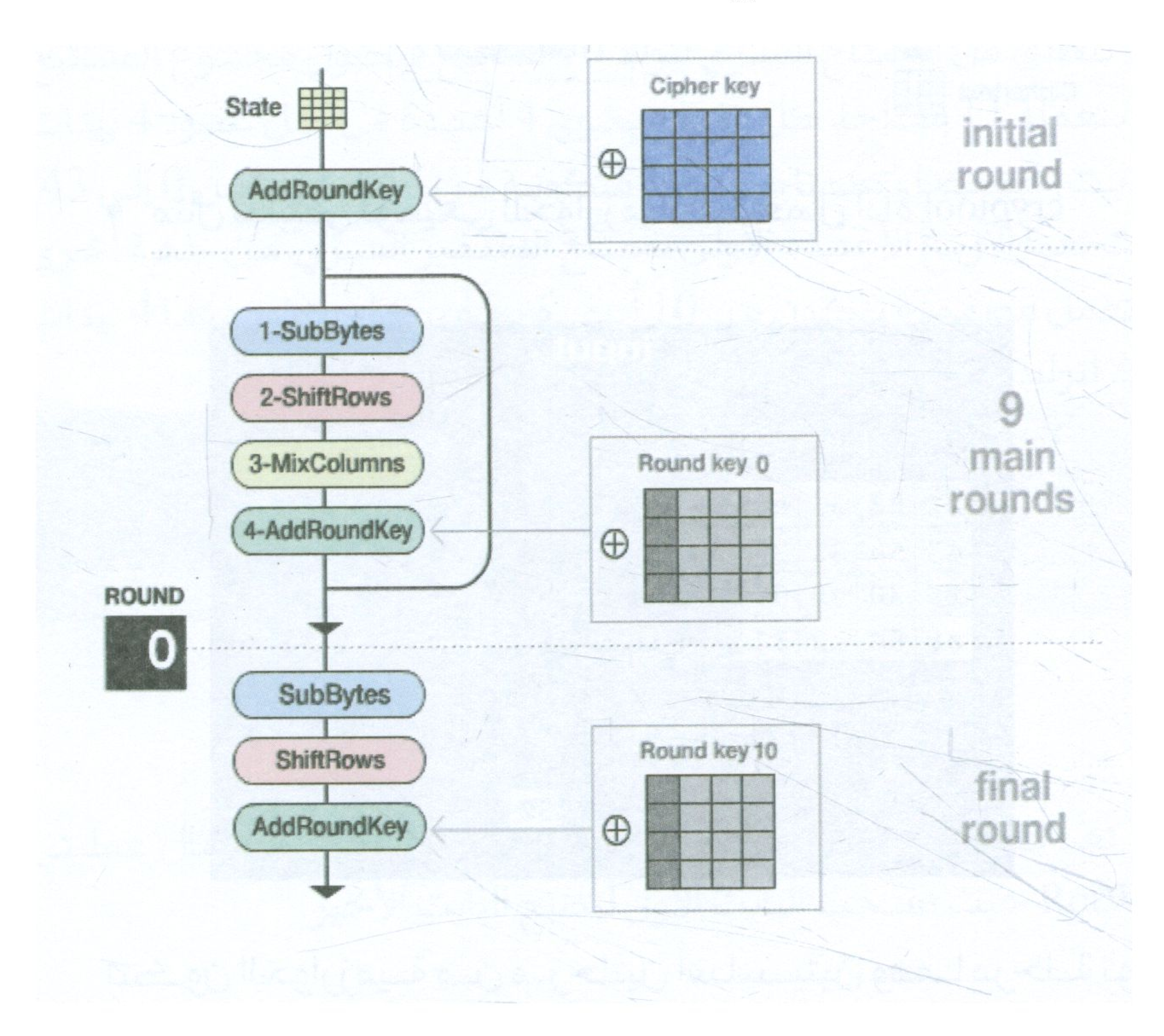
تتفادى هذه الطريقة بعض المشاكل في نظام الترميز الإليكتروني، إذ أنه حتى لو تم استخدام نفس الكتلة النصية في نظام الترميز الاليكتروني في الترميز النصي فإن ذلك لا ينجم عنه تكرار في الترميز النصي المشفر. ومن خصائصها أن كل كتلة نصية مشفرة مربوطة بكل الكتل المشفرة لها، ولكن إذا ضاعت كتلة نصية مشفرة فإن التأثير لا يطال إلا الكتلة النصية المفترض توليدها من الكتلة الضائعة، وكذلك الكتلة الموالية لها، وبقية النص لا تتأثر بهذا الضياع.

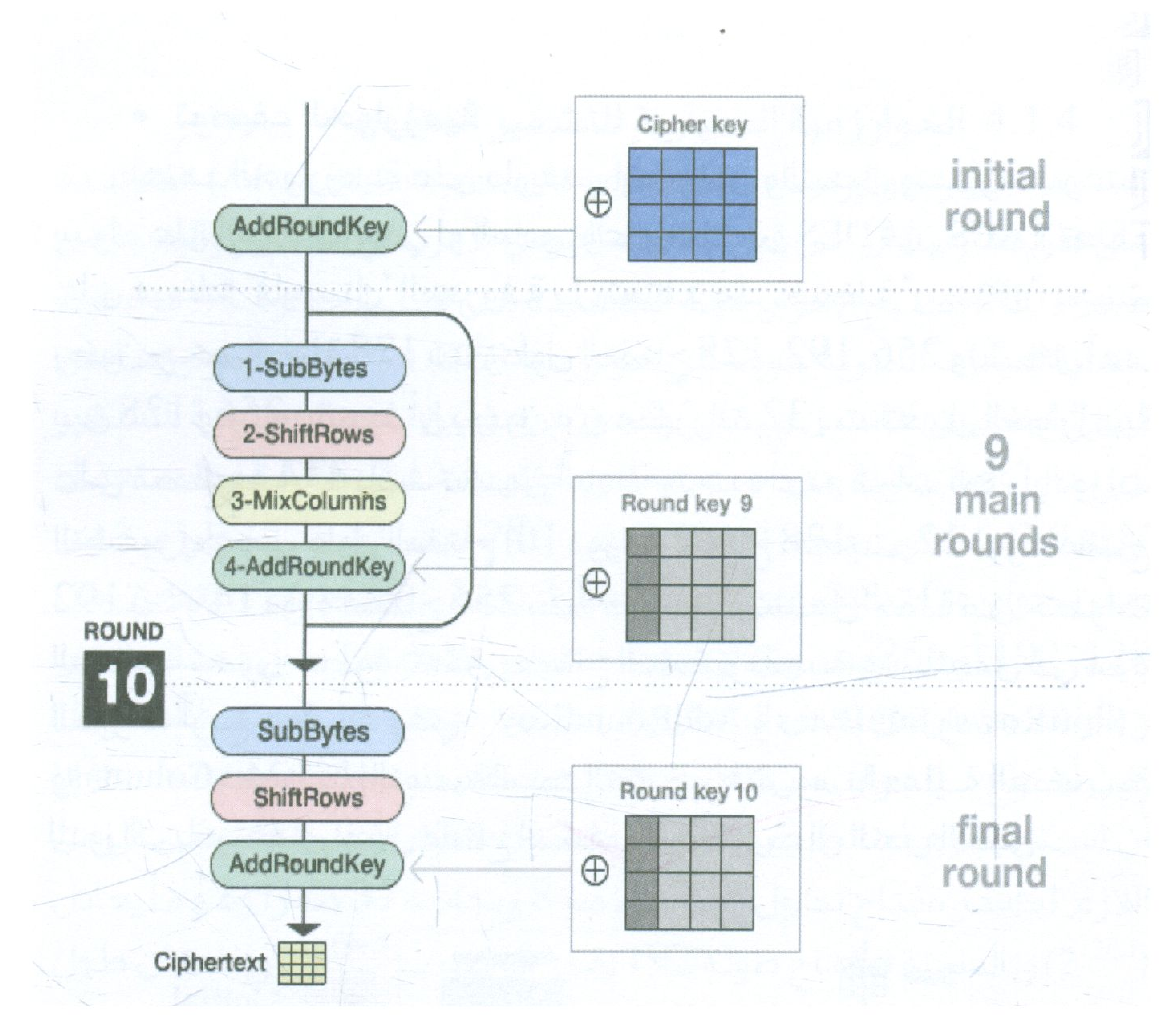
4.1.4 الخوارزمية المعيارية للتشفير المتقدم

تم اختيار خوارزمية AES (Advanced Encryption Standard) المعتمدة على طريقة "ريينداك" من قبل المعهد القومي الأمريكي للمعايير والتقنية في عام 2001 لتصبح الخوارزمية المعتمدة من قبل الحكومة الأمريكية لحماية البيانات السرية عوضا عن خوارزمية DES. تم تطوير الخوارزمية من قبل البلجكيين "فانسن ريمان" و"جون ديمن" باستعمال طريقة التشفير "ريينداك" المقتبسة من اسميهما. وبأكثر دقة فإن AES هي استعمال "ريينداك" لما يكون حجم الكتلة محددا ب128 بت. تصنف الخوارزمية من ضمن خوارزميات التشفير التماثلي حيث تعمل بمفتاح سري يتكون من 128, 192 أو 256 بت، وتزداد جودة التشفير كلما كان طول المفتاح أكبر. و بالرغم من توصل بعض الباحثين إلى طريقة تُمكن من كسر AES خلال زمن أسرع بـ 4 مرات من الطرق التقليدية كهجوم (Brute-force attack) إلا أن استخدام هذه الطريقة لا يعني سقوط AES الآن، حيث يبقى الزمن اللازم لكسر مفتاح بطول 128 بت هو 8 وبجانبه 37 صفرا وهو ما يعادل ($2^{126.1}$) وبالنسبة لمفتاح طوله 192 يقدر الوقت ب $2^{189.7}$ وإذا كان طول المفتاح 256 فيقدر الوقت ب $2^{254.4}$، إلا أن السرعات الأعلى للحواسيب في المستقبل قد تسبب سقوط AES. بخلاف أنواع الهجمات السابقة على AES، فإن الهجوم الجديد يعمل مع أي مفتاح خاص عشوائي ويستخدم أسلوباً شائعاً للهجمات ضد خوارزميات التشفير يسمّى المقابلة في المنتصف (أو الوسط) (meet-in-the-middle attack). بشكل أساسي، يكون لدى المهاجم قائمة لنصوص غير مشفرة (plaintext) ونصوصها المقابلة المشفّرة (ciphertext) والتي يقوم بتشفيرها وفك تشفيرها ليرى إن كانت سـ "تتقابل" في المنتصف. استطاع الباحثون تسريع هذه العملية من خلال استخدام أسلوب جديد أطلقوا عليه اسم "biclique" ، مكنّهم من التخلص من بعض الوقت اللازم لتنفيذ الإجراءات التقليدية. أما إذا استعملنا أسلوب "Related-key Attaks" فإن الوقت اللازم ينخفض إلى 2^{176} و $2^{99.5}$ بالنسبة لمفتاحي 192 و256.

• **توصيف الخوارزمية**

تعتمد الخوارزمية على طريقتي التعويض والتبديل وتمتاز بسرعتها سواء على العتاد البرمجي أو المادي. وعلى عكس DES فإن AES لا تعتمد على هيكلة "فايستل" المعروفة بل تعتمد على هيكلة "رييندال" بحيث يكون حجم الكتلة 128 بت وطول المفتاح 128، 192، 256 وبشكل أعم بين 128 و256 بشرط أن يكون من مكررات 32 بت. تعمل الخوارزمية على مصفوفة 4*4 بايت تسمى "state" وتحدد عدد مرات تكرار دورات التشفير بحسب طول المفتاح (10 دورات لمفتاح 128بت، 12 دورة لمفتاح 192 بت و 14 دورة لمفتاح 256 بت). كل دورة تشمل جملة من خطوات المعالجة تحوي خطوة تتعلق بمفتاح التشفير نفسه. ونستعمل في هذه الدورات أربع عمليات وهي AddRoundKey و SubBytes وShiftRows وMixColumns. أما بالنسبة لفك التشفير فنقوم بالعملية العكسية للدورات باستعمال نفس مفتاح التشفير للحصول على النص الأصلي.





• مثال تطبيقي توضيحي للخوارزمية باستعمال أداة cryptool

Input

State

32	88	31	e0
43	5a	31	37
f6	30	98	07
a8	8d	a2	34

This is a block from the plaintext message to be encrypted.

Cipher key

2b	28	ab	09
7e	ae	f7	cf
15	d2	15	4f
16	a6	88	3c

Hexadecimal notation (sample):

32 = 00110010 (1 byte)
3hex 2hex

تتكـون الخـوارزميـة مـن مـرحلتين أسـاسـيتين وهمـا مرحلـة توليـد المفاتيح ومرحلة التشفير:

State

32	88	31	e0
43	5a	31	37
f6	30	98	07
a8	8d	a2	34

to Encryption Process

Ⓐ

Cipher key

2b	28	ab	09
7e	ae	f7	cf
15	d2	15	4f
16	a6	88	3c

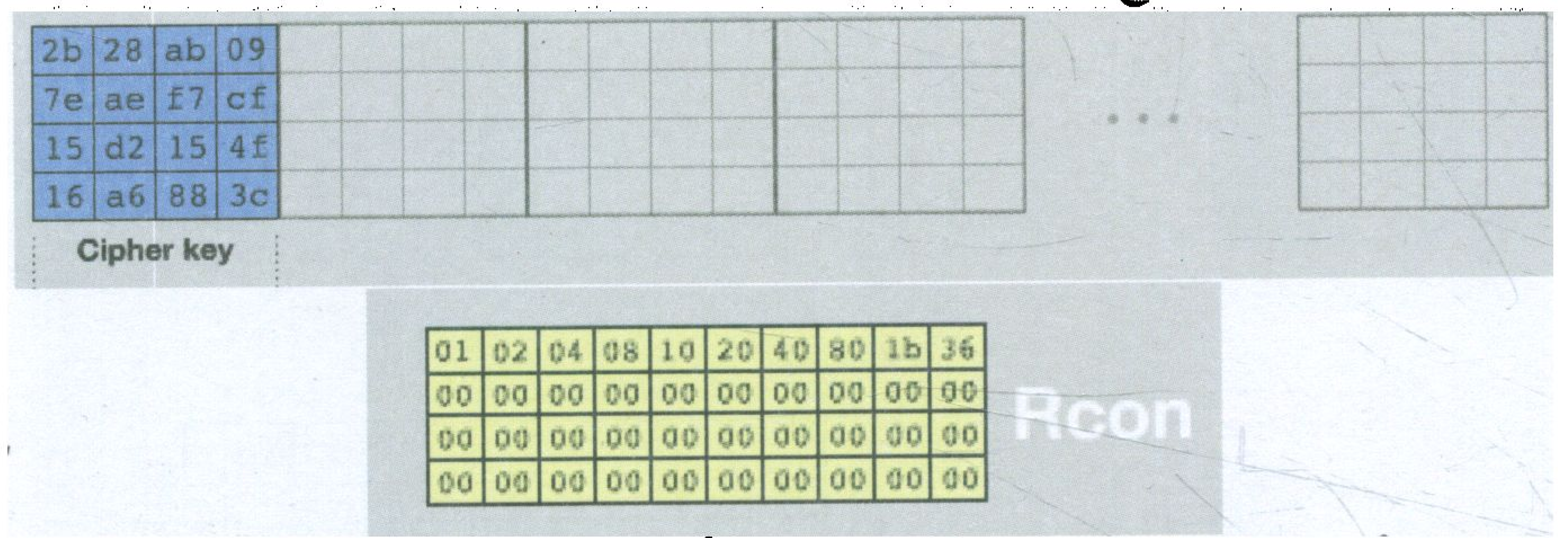
to Key Schedule

Ⓑ

مرحلة توليد المفاتيح

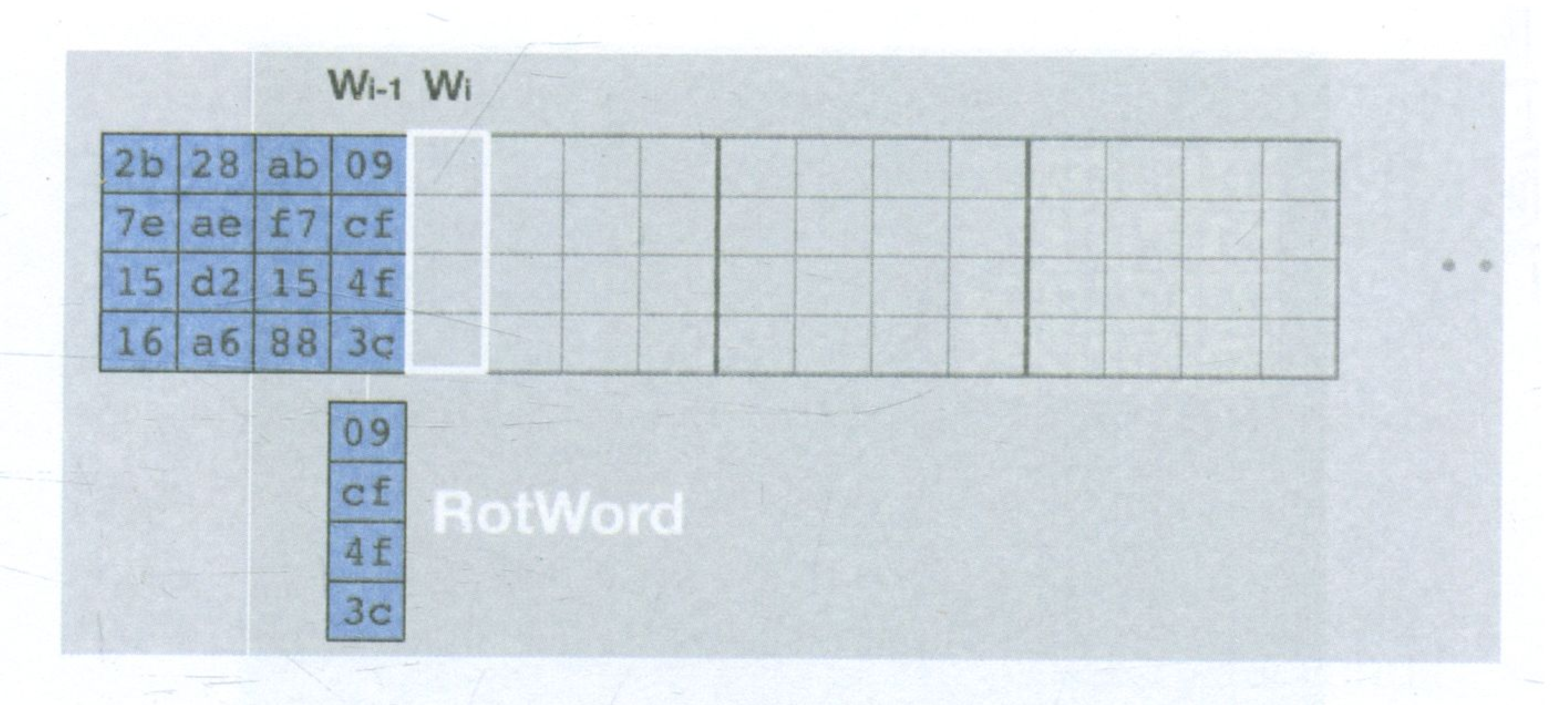
يقع استخراج مفاتيح الدورات من مفتاح التشفير السري حسب جدولة مخصوصة تعرف ب"جدول مفتاح رييندال" وهذه الخطوة تعرف ب "key expansion" وفيما يلي تفصيل مراحلها:

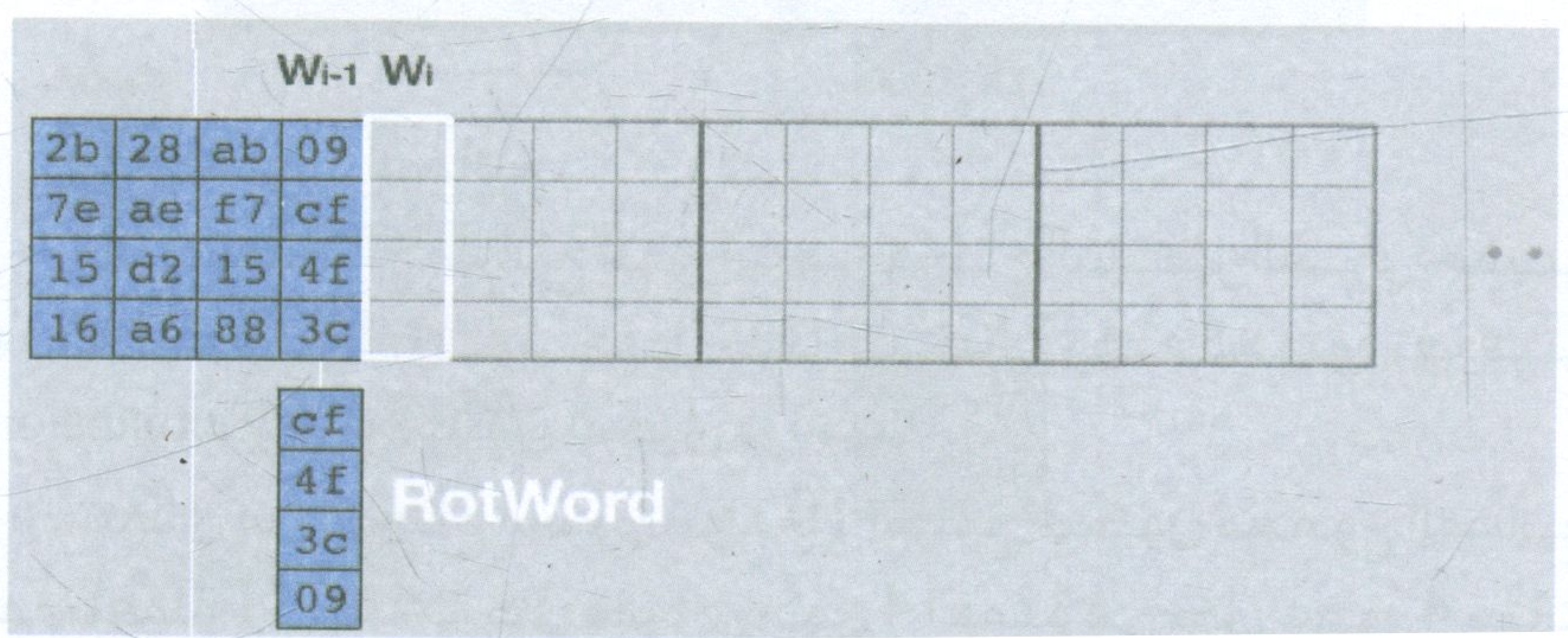
ننطلق من المفتاح السري لنولد 10 مفاتيح فيكون مجموع المفاتيح المستعملة 11 مفتاحا. كل مفتاح يحوي 4 أعمدة في كل عمود 4 بيتات (32 بت) فيتحصل عندنا مصفوفة متكونة من 44 عمود (من 0 إلى 43) بحيث تعبأ الأربع الأعمدة الأولى بمفتاح التشفير السري. من جهة أخرى نستعمل مصفوفة تتكون من 10 أعمدة يحوي كل عمود منها 4 بيتات وذلك لتوليد المفاتيح العشرة وهذه المصفوفة تسمى Rcon



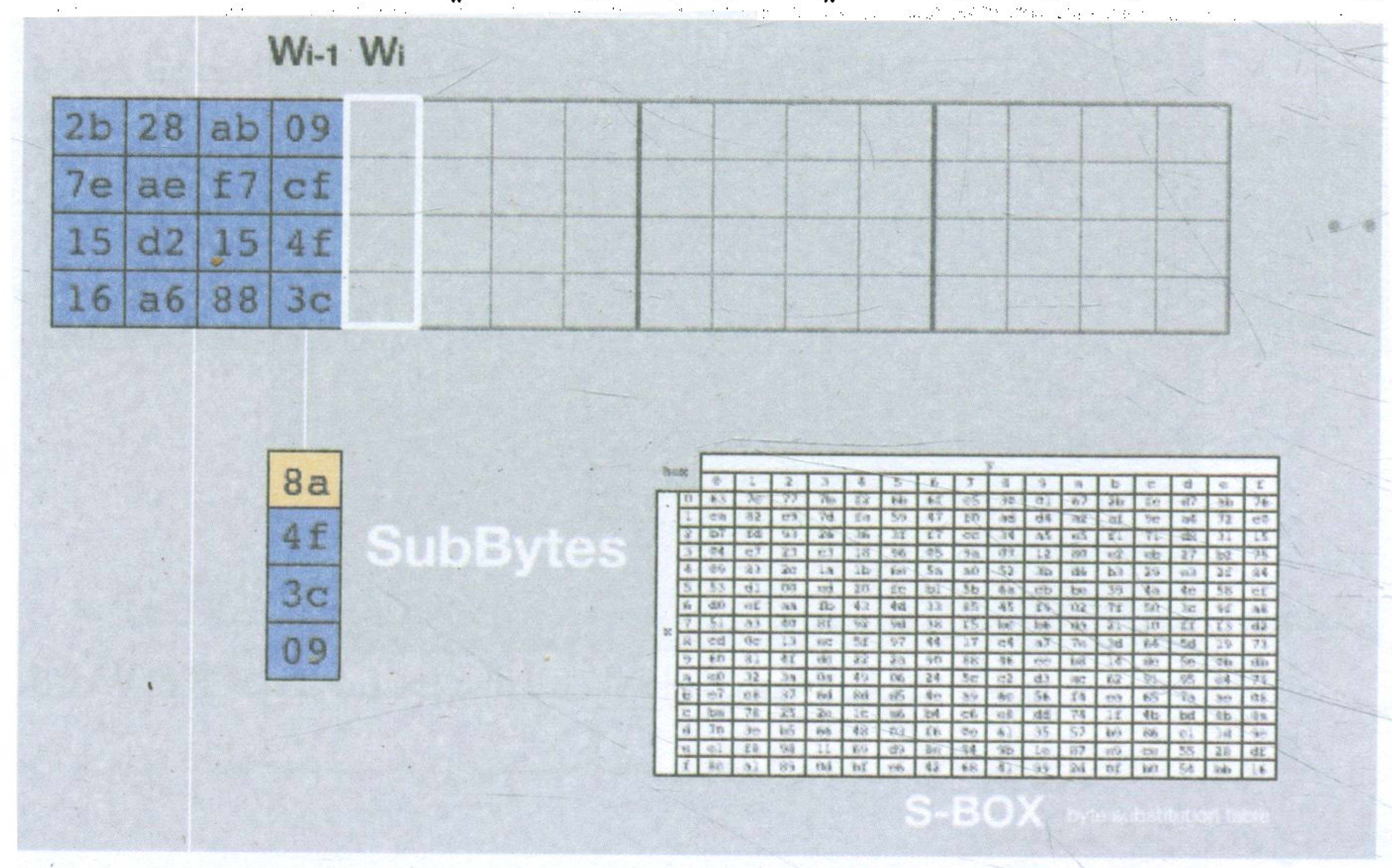
01	02	04	08	10	20	40	80	1b	36
00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00

تعبأ الكلمات w40 وw8 و مكررات أربعة إلى w40 باستعمال عمليتي RotWord حيث نسحب البايت الأول ليصبح البايت الأخير

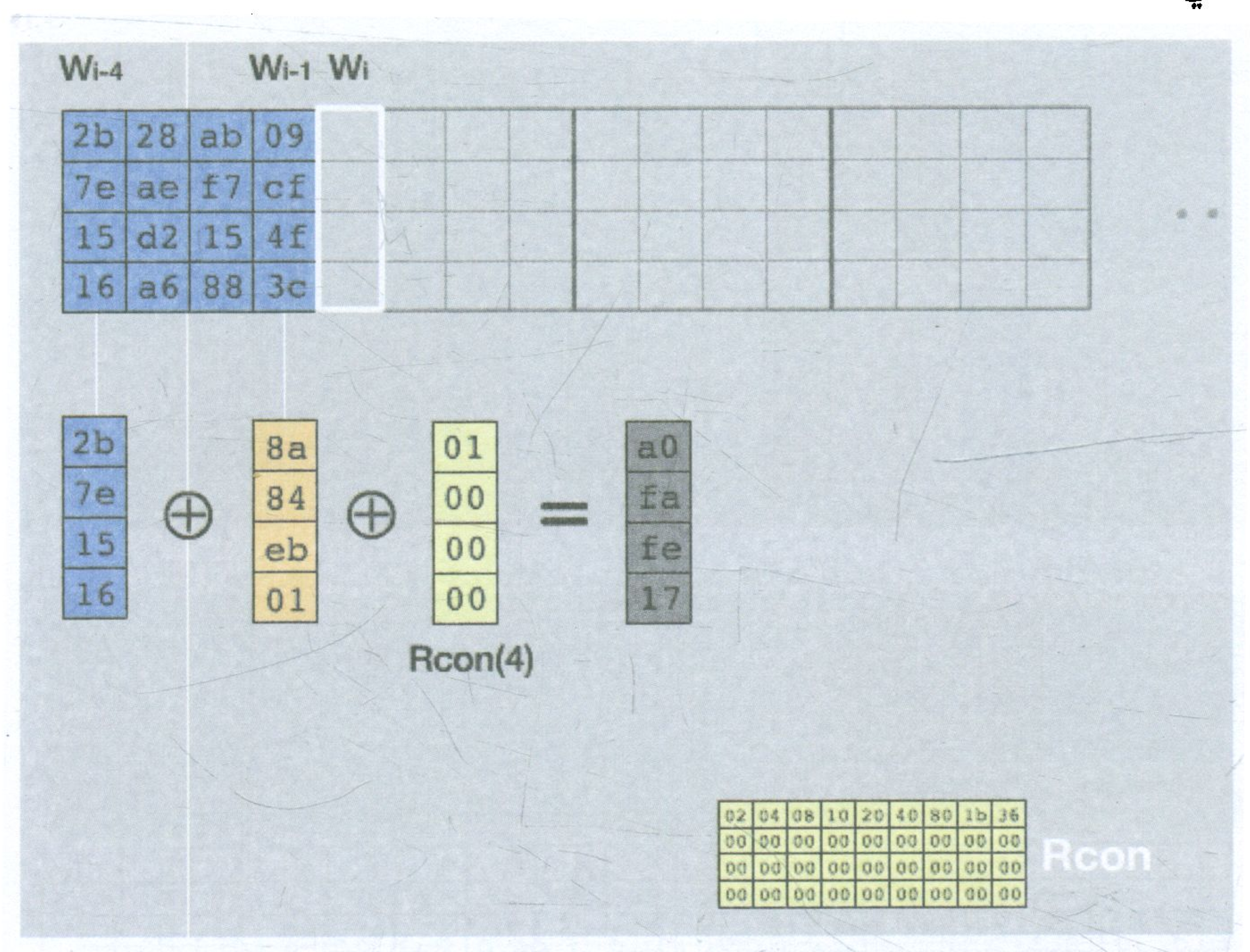




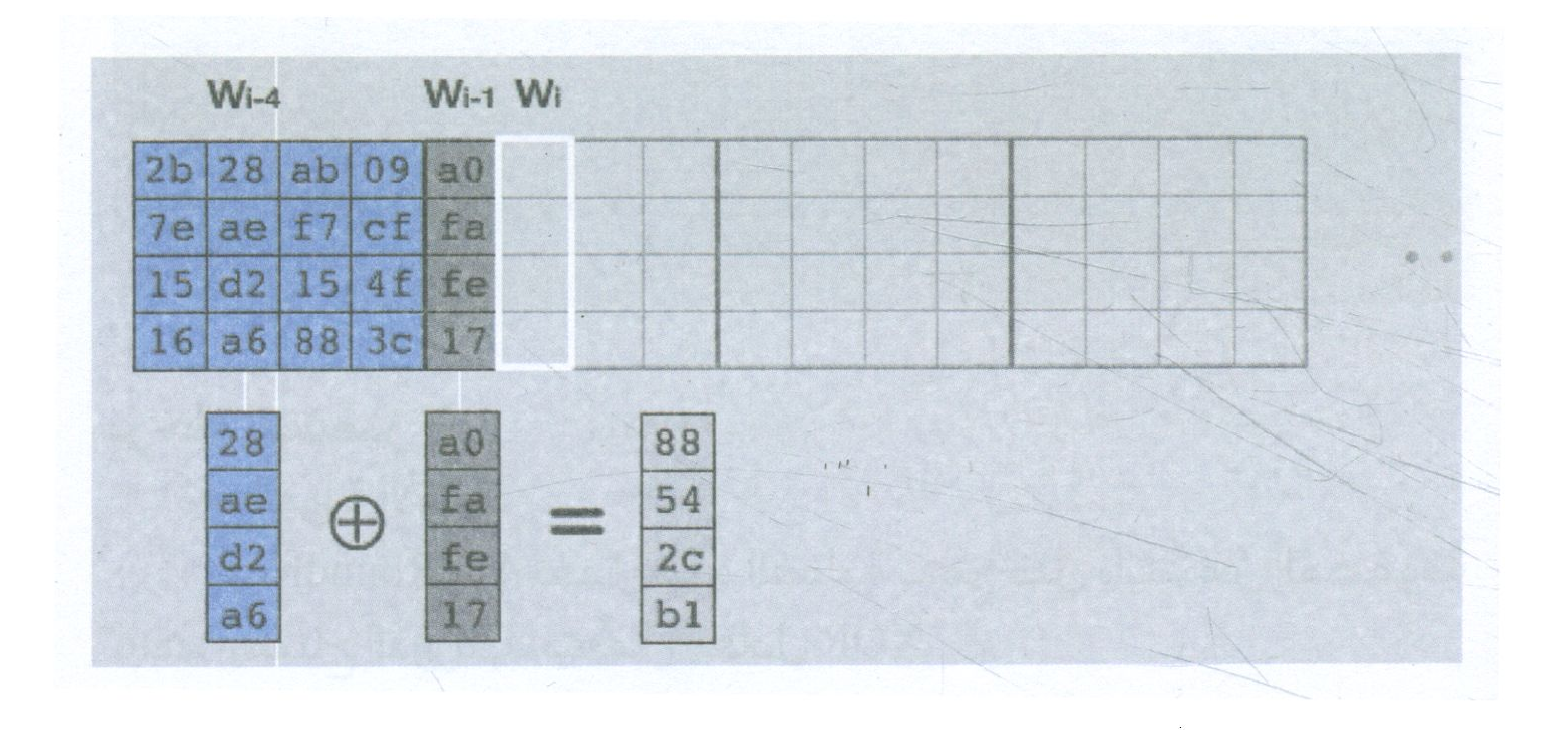
نقوم بعملية التعويض باستعمال جدول التعويض بحيث نستخرج القيمة من الجدول باعتماد احداثية السطر هو الحرف الأول من البايت وإحداثية العمود هو الحرف الثاني ونقوم بتعويضها في الكلمة.



نقـوم بعمليـة xor بـين الكلمـة الأولى والمتحصل عليـه من العمليـة السابقة و العمود الأول من Rcon لكي نحصل على أول عمـود من المفتاح الثاني.



نقـوم بعمليـة xor بين العمـود الثاني مـن المفتـاح السـري مـع العمـود المتحصل عليه آنفا للحصول على العمود الثاني للمفتاح الثاني



نقوم بنفس العملية لكي نحصل على بقية أعمدة المفتاح الثاني

			W_{i-4}			W_{i-1}	W_i
2b	28	ab	09	a0	88	23	
7e	ae	f7	cf	fa	54	a3	
15	d2	15	4f	fe	2c	39	
16	a6	88	3c	17	b1	39	

$$\begin{bmatrix} 09 \\ cf \\ 4f \\ 3c \end{bmatrix} \oplus \begin{bmatrix} 23 \\ a3 \\ 39 \\ 39 \end{bmatrix} = \begin{bmatrix} 2a \\ 6c \\ 76 \\ 05 \end{bmatrix}$$

Cipher key				Round key 1			
2b	28	ab	09	a0	88	23	2a
7e	ae	f7	cf	fa	54	a3	6c
15	d2	15	4f	fe	2c	39	76
16	a6	88	3c	17	b1	39	05

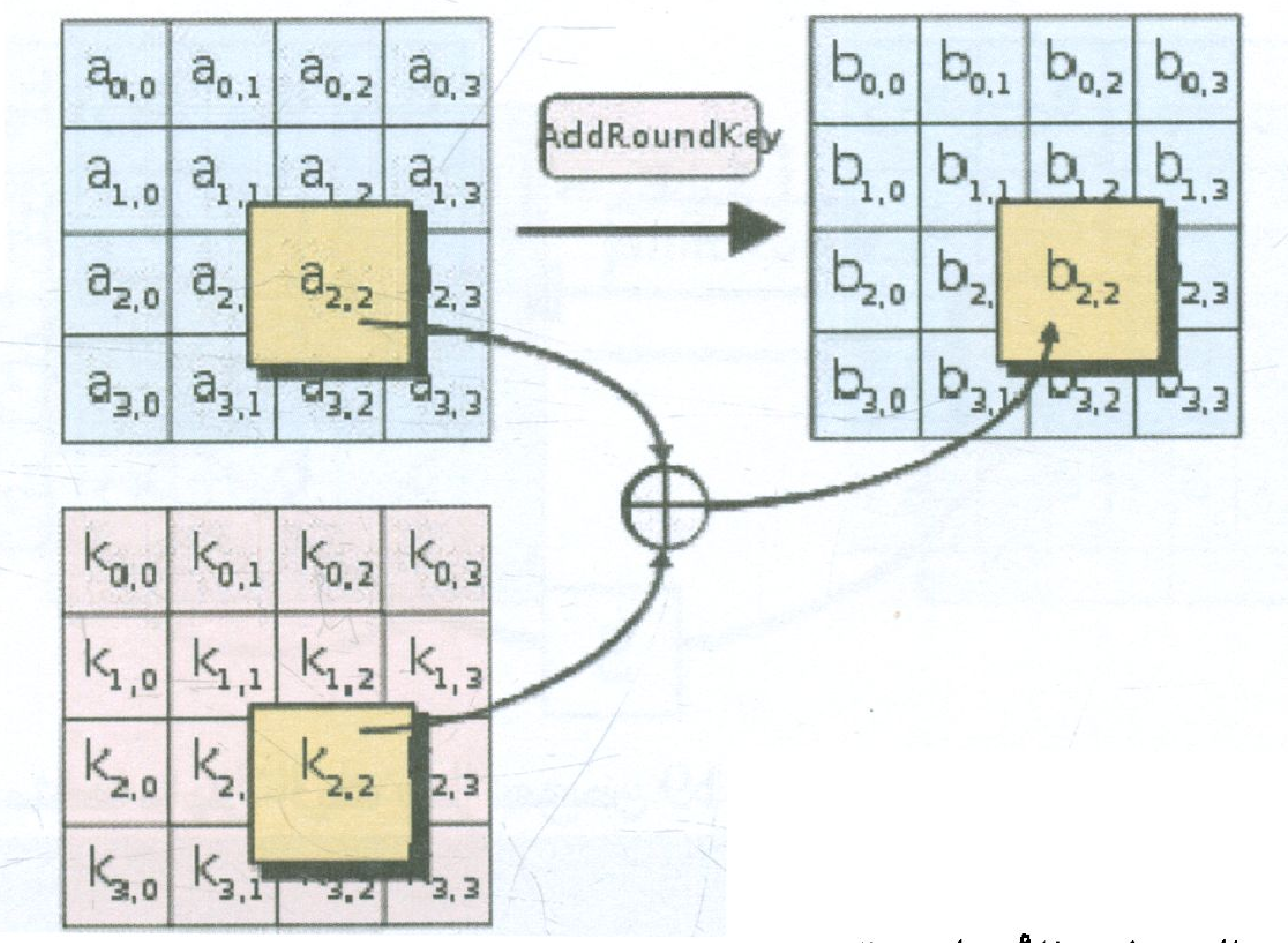
لتوليد بقية المفاتيح نقوم بنفس العمليات السابقة تماما وبنفس الترتيب إلا أننا نستعمل في كل مرة المفتاح السابق مباشرة لنا مع العمود الموافق في Rcon لنحصل في النهاية على المفاتيح كلها.

Cipher key				Round key 1				Round key 2				Round key 3				...	Round key 10			
2b	28	ab	09	a0	88	23	2a	f2	7a	59	73	3d	47	1e	6d	...	d0	c9	e1	b6
7e	ae	f7	cf	fa	54	a3	6c	c2	96	35	59	80	16	23	7a	...	14	ee	3f	63
15	d2	15	4f	fe	2c	39	76	95	b9	80	f6	47	fe	7e	88	...	f9	25	0c	0c
16	a6	88	3c	17	b1	39	05	f2	43	7a	7f	7d	3e	44	3b	...	a8	89	c8	a6

مرحلة التشفير

- الدورة الأولية

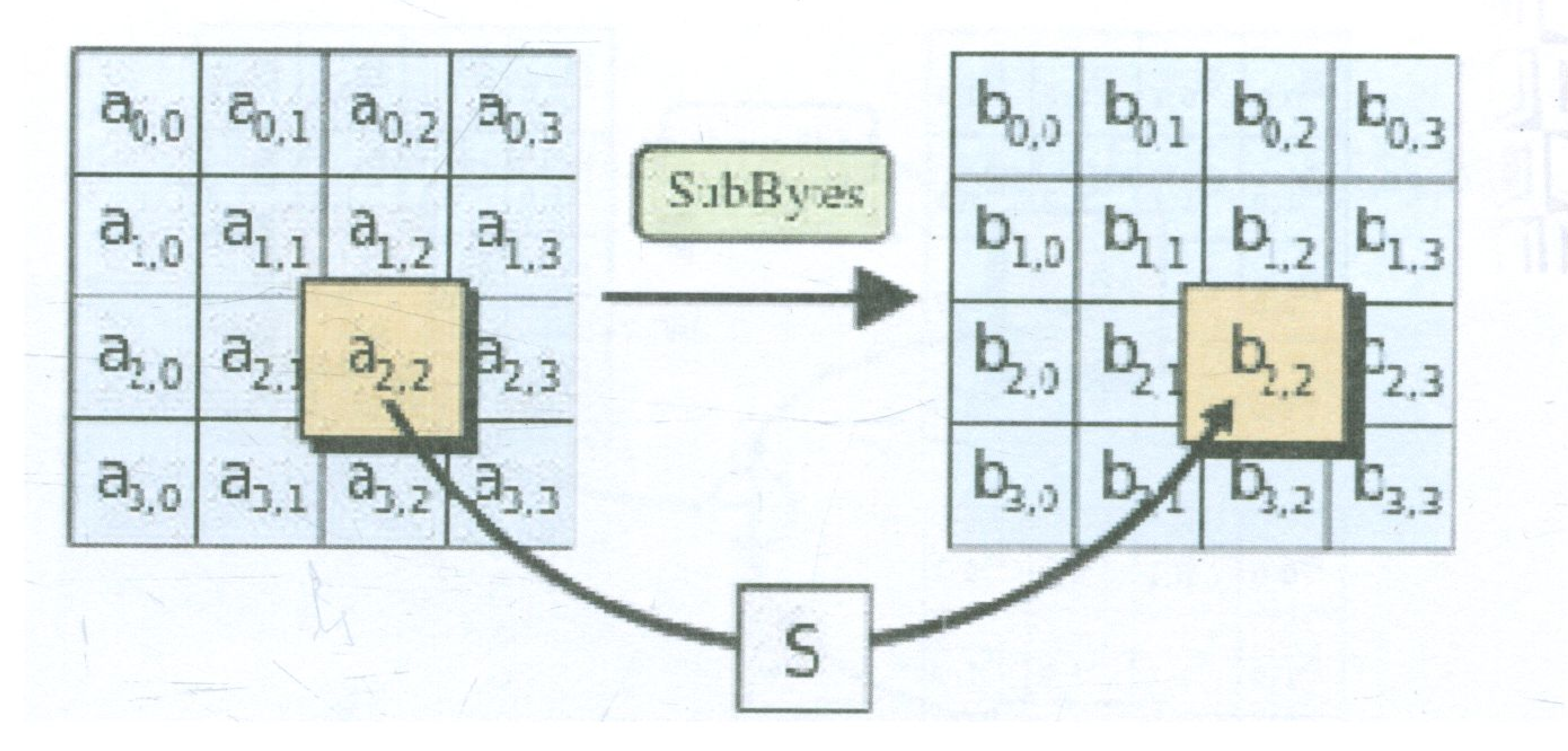
AddRoundKey تعمل هذه العملية بدمج كل بايت من المصفوفة "state" بمفتاح الدورة باستعمال معامل XOR



- **الدورات الأساسية**

SubBytes وهي عملية تعويض غير خطية حيث تعوض كل بايت بأخرى باستعمال جدول مخصص للتعويض بنفس طريقة التعويض المشروحة آنفا في توليد المفاتيح

		y															
		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
x	0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
	1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
	2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
	3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
	4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
	5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
	6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
	7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
	8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
	9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
	a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
	b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
	c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
	d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
	e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
	f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16



ومثالها ما يلي حيث يتم تعويض 19 ب d4

Round 1

19

	a0	9a	e9
3d	f4	c6	f8
e3	e2	8d	48
be	2b	2a	08

d4

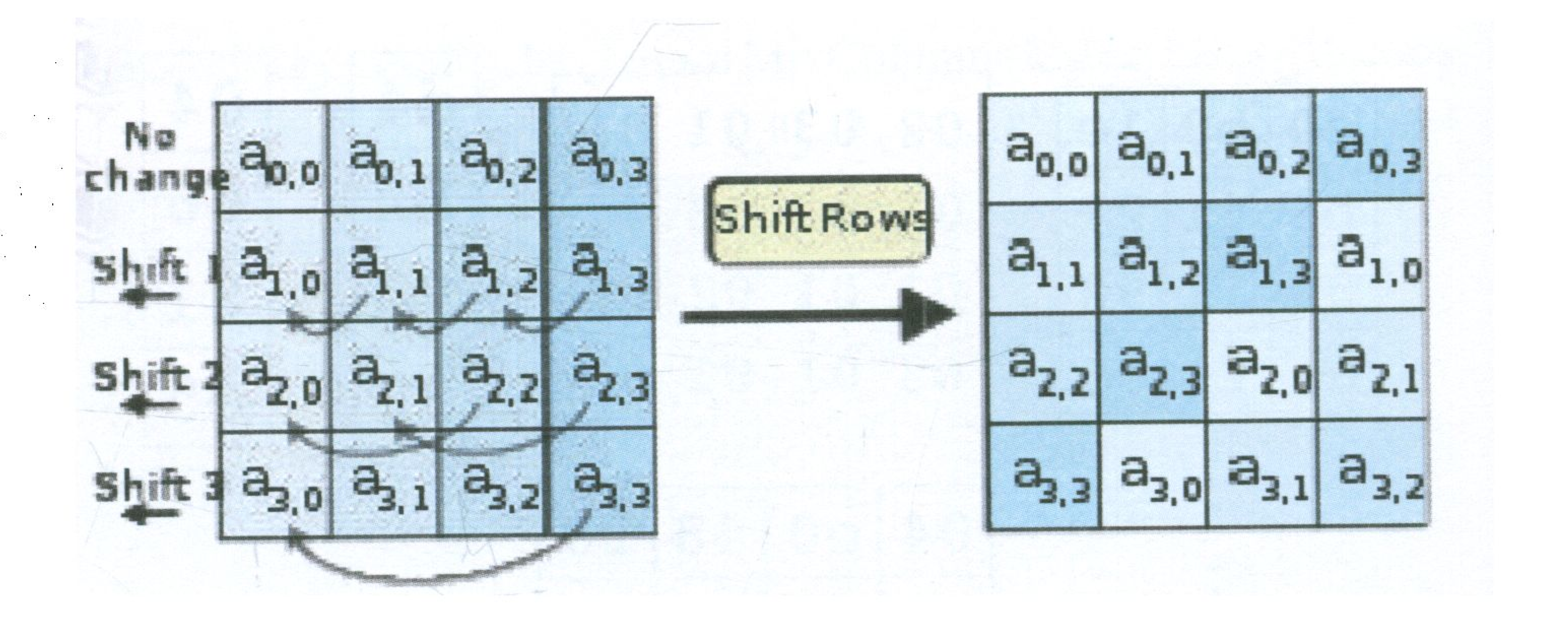
hex x\y	0	1	2	3	4	5	6	7				b	c	d	e	f
0	63	7c	77	7b	f2	6b	6f	c5				2b	fe	d7	ab	76
1	ca	82	c9	7d	fa	59	47	f0				af	9c	a4	72	c0
2	b7	fd	93	26	36	3f	f7	cc				f1	71	d8	31	15
3	04	c7	23	c3	18	96	05	9a				e2	eb	27	b2	75
4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

S-BOX byte substitution table

فيتحصل عندنا في النهاية المصفوفة التالية

d4	e0	b8	1e
27	bf	b4	41
11	98	5d	52
ae	f1	e5	30

ShiftRows وهي عملية تغيير لأمكنة الأسطر بحيث يسحب كل سطر دوريا بدرجة سحب معينة في المصفوفة "state".



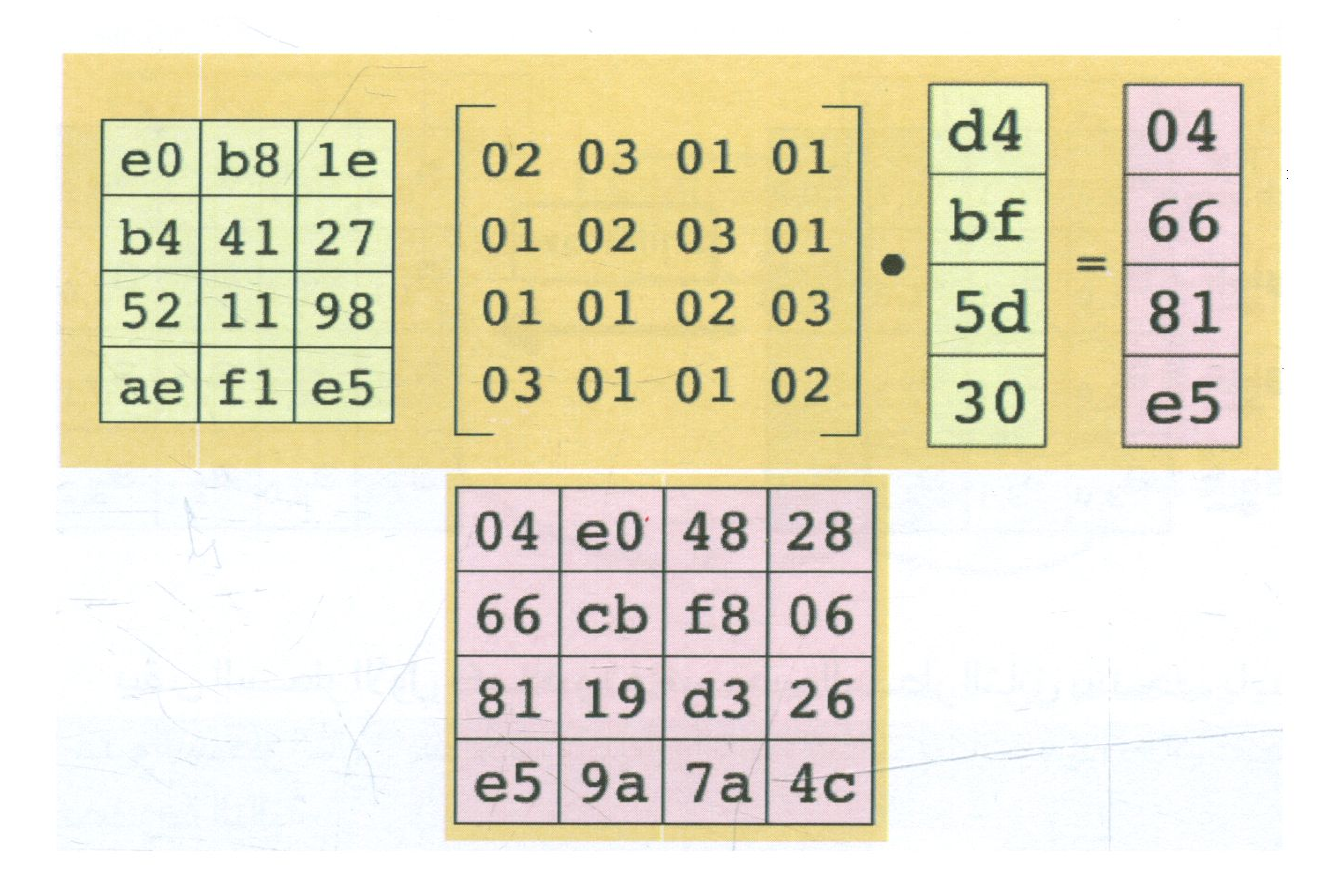
يبقى السطر الأول كما هو ثم نسحب السطر الثاني بسحب بايت واحد والسطر الثالث بسحب بايتين والرابع بثلاث بيتات. فإذا كان عندنا المصفوفة التالية

d4	e0	b8	1e
27	bf	b4	41
11	98	5d	52
ae	f1	e5	30

فسنحصل على المصفوفة الموالية بعد عمليات السحب

d4	e0	b8	1e
bf	b4	41	27
5d	52	11	98
30	ae	f1	e5

MixColumns وهي عملية خلط في أعمدة المصفوفة بدمج الأربع بيتات لكل عمود مضروبة دوريا على حقل جالوا "galois field" لريندال باستعمال مصفوفة محددة. وتعتبر هذه العملية هي العملية الرئيسية لإحداث الفوضى والانتشار في البيانات مع عملية ShiftRows



AddRoundKey وفي هذه العملية نقوم بعملية xor بين كلمن أعمدة مفتاح الدورة وأعمدة مصفوفة البيانات لنحصل على المصفوفة المشفرة للدورة الأولى

e0	48	28
cb	f8	06
19	d3	26
9a	7a	4c

$$\begin{bmatrix}04\\66\\81\\e5\end{bmatrix} \oplus \begin{bmatrix}a0\\fa\\fe\\17\end{bmatrix} = \begin{bmatrix}a4\\9c\\7f\\f2\end{bmatrix}$$

88	23	2a
54	a3	6c
2c	39	76
b1	39	05

a4	68	6b	02
9c	9f	5b	6a
7f	35	ea	50
f2	2b	43	49

نقوم بنفس العمليات والمراحل للتسع الدورات الباقية إلا أن الدورة النهائية تحتوي فقط على SubBytes و ShiftRows و AddRoundKey

ونسـتثني منهـا عمليـة MixColumns لنحصـل علـى المصـفوفة المشـفرة أخيرا كما هو مبين في الشكل الموالي

	Start of round	After SubBytes	After ShiftRows	After MixColumns		Round key	
Input	32 88 31 e0					2b 28 ab 09	
	43 5a 31 37				⊕	7e ae f7 cf	=
	f6 30 98 07					15 d2 15 4f	
	a8 8d a2 34					16 a6 88 3c	
Round 1	19 a0 9a e9	d4 e0 b8 1e	d4 e0 b8 1e	04 e0 48 28		a0 88 23 2a	
	3d f4 c6 f8	27 bf b4 41	bf b4 41 27	66 cb f8 06	⊕	fa 54 a3 6c	=
	e3 e2 8d 48	11 98 5d 52	5d 52 11 98	81 19 d3 26		fe 2c 39 76	
	be 2b 2a 08	ae f1 e5 30	30 ae f1 e5	e5 9a 7a 4c		17 b1 39 05	
Round 2	a4 68 6b 02	49 45 7f 77	49 45 7f 77	58 1b db 1b		f2 7a 59 73	
	9c 9f 5b 6a	de db 39 02	db 39 02 de	4d 4b e7 6b	⊕	c2 96 35 59	=
	7f 35 ea 50	d2 96 87 53	87 53 d2 96	ca 5a ca b0		95 b9 80 f6	
	f2 2b 43 49	89 f1 1a 3b	3b 89 f1 1a	f1 ac a8 e5		f2 43 7a 7f	
Round 3	aa 61 82 68	ac ef 13 45	ac ef 13 45	75 20 53 bb		3d 47 1e 6d	
	8f dd d2 32	73 c1 b5 23	c1 b5 23 73	ec 0b c0 25	⊕	80 16 23 7a	=
	5f e3 4a 46	cf 11 d6 5a	d6 5a cf 11	09 63 cf d0		47 fe 7e 88	
	03 ef d2 9a	7b df b5 b8	b8 7b df b5	93 33 7c dc		7d 3e 44 3b	
Round 4	48 67 4d d6	52 85 e3 f6	52 85 e3 f6	0f 60 6f 5e		ef a8 b6 db	
	6c 1d e3 5f	50 a4 11 cf	a4 11 cf 50	d6 31 c0 b3	⊕	44 52 71 0b	=
	4e 9d b1 58	2f 5e c8 6a	c8 6a 2f 5e	da 38 10 13		a5 5b 25 ad	
	ee 0d 38 e7	28 d7 07 94	94 28 d7 07	a9 bf 6b 01		41 7f 3b 00	
Round 5	e0 c8 d9 85	e1 e8 35 97	e1 e8 35 97	25 bd b6 4c		d4 7c ca 11	
	92 63 b1 b8	4f fb c8 6c	fb c8 6c 4f	d1 11 3a 4c	⊕	d1 83 f2 f9	=
	7f 63 35 be	d2 fb 96 ae	96 ae d2 fb	a9 d1 33 c0		c6 9d b8 15	
	e8 c0 50 01	9b ba 53 7c	7c 9b ba 53	ad 68 8e b0		f8 87 bc bc	

	Start of round	After SubBytes	After ShiftRows	After MixColumns		Round key	
Round 6	f1 c1 7c 5d	a1 78 10 4c	a1 78 10 4c	4b 2c 33 37		6d 11 db ca	
	00 92 c8 b5	63 4f e8 d5	4f e8 d5 63	86 4a 9d d2	⊕	88 0b f9 00	=
	6f 4c 8b d5	a8 29 3d 03	3d 03 a8 29	8d 89 f4 18		a3 3e 86 93	
	55 ef 32 0c	fc df 23 fe	fe fc df 23	6d 80 e8 d8		7a fd 41 fd	
Round 7	26 3d e8 fd	f7 27 9b 54	f7 27 9b 54	14 46 27 34		4e 5f 84 4e	
	0e 41 64 d2	ab 83 43 b5	83 43 b5 ab	15 16 46 2a	⊕	54 5f a6 a6	=
	2e b7 72 8b	31 a9 40 3d	40 3d 31 a9	b5 15 56 d8		f7 c9 4f dc	
	17 7d a9 25	f0 ff d3 3f	3f f0 ff d3	bf ec d7 43		0e f3 b2 4f	
Round 8	5a 19 a3 7a	be d4 0a da	be d4 0a da	00 b1 54 fa		ea b5 31 7f	
	41 49 e0 8c	83 3b e1 64	3b e1 64 83	51 c8 76 1b	⊕	d2 8d 2b 8d	=
	42 dc 19 04	2c 86 d4 f2	d4 f2 2c 86	2f 89 6d 99		73 ba f5 29	
	b1 1f 65 0c	c8 c0 4d fe	fe c8 c0 4d	d1 ff cd ea		21 d2 60 2f	
Round 9	ea 04 65 85	87 f2 4d 97	87 f2 4d 97	47 40 a3 4c		ac 19 28 57	
	83 45 5d 96	ec 6e 4c 90	6e 4c 90 ec	37 d4 70 9f	⊕	77 fa d1 5c	=
	5c 33 98 b0	4a c3 46 e7	46 e7 4a c3	94 e4 3a 42		66 dc 29 00	
	f0 2d ad c5	8c d8 95 a6	a6 8c d8 95	ed a5 a6 bc		f3 21 41 6e	
Round 10	eb 59 8b 1b	e9 cb 3d af	e9 cb 3d af			d0 c9 e1 b6	
	40 2e a1 c3	09 31 32 2e	31 32 2e 09		⊕	14 ee 3f 63	=
	f2 38 13 42	89 07 7d 2c	7d 2c 89 07			f9 25 0c 0c	
	1e 84 e7 d2	72 5f 94 b5	b5 72 5f 94			a8 89 c8 a6	
Output	39 02 dc 19						
	25 dc 11 6a						
	84 09 85 0b						
	1d fb 97 32						

4.2 التشفير بالمفتاح العام أو التشفير غير التناظري

يعاني نظام التشفير التماثلي من مشكلة تبادل المفاتيح السرية بين المتخاطبين على الشبكة، نظرًا لكونها لا بد أن تبقى معلومة من طرف هؤلاء المتخاطبين فقط. ولهذا جاء التشفير بالمفتاح العام حلاً لهذه المشكلة؛ إذ لم يعد هناك حاجة لأن يكون المفتاح مشتركًا بين أكثر من طرف. هذا بدوره يحل مشكلة أخرى في التشفير التماثلي، وهي مشكلة إنكار الإرسال، فمثلا لو أن زيدًا وعبيدًا يملكان نفس المفتاح السري في التشفير التناظري فإنه لو فرضنا أن زيدًا أرسل رسالة ما إلى عبيد، ثم أنكر إرسال الرسالة فإن عبيدًا وإن كان متأكدًا من إرسال زيد له لا يمكن أن يثبت ذلك، لأنه في كل الأحوال محل تهمة؛ لأنه يملك نفس المفتاح فقد يقال له: أنت الذي شفرت الرسالة بالمفتاح المشترك بينكما ثم أرسلت الرسالة باسم زيد لنفسك. هجوم الإنكار هذا انتهى في التشفير بالمفتاح العام، لأنه ليس ثمة اشتراك في المفتاح الخاص، فكل مفتاح خاص يملكه شخص واحد، وشخص واحد فقط. كذلك من مزايا التشفير بالمفتاح العام خدمة التوقيع الالكتروني، لأن المفتاح الخاص يمكن أن يعتبر بصمة خاصة بكل شخص، فهي تُعبِّر عن هويته، وهناك المزيد من الخدمات الأخرى التي سنعرض لها في هذا الكتاب.

4.2.1 التمثيل الرياضي

تعتبر $f: X \to Y$ دالة ذات اتجاه واحد فيما إذا كنت f سهلة في حسابها لكل $x \in X$ ولكن f^{-1} صعب حسابها.

مثال باقي قسمة الجذور التكعيبية:

لنحدد $p = 48611$ و $q = 53993$ ولتكن

$n = p\ q = 2624653723$ و $X = \{1, 2, \dots, n-1\}$

لتكن $f: X \to N; f(x) = x^3 mod\ n$

مثال: فحساب $f(2489991) = 1981394214$ سهل، ولكن حساب عكسها صعب، أي البحث على x باقي قسمة تكعيبه على n يساوي 1981394214

تعتبر دالة الاتجاه الواحد $f: X \to Y$ ذات باب خلفي فيما إذا توفر لدينا معلومة إضافية تجعل من الممكن إيجاد لكل $y \in Im(f)$ قيمة x بحيث تكون $f(x) = y$

مثال: باقي قسمة الجذور التكعيبية سهل لو علمنا p و q.

لتكن $\{E_e: e \in K\}$ و $\{D_e: e \in K\}$ لنظام تشفير معين، لنعتبر أزواج التشفير (E_e, D_d) حيث إذا علمنا قيمة E_e فليس بالممكن ولو عرفنا $c \in C$ أن نحصل على $m \in M$ حيث تكون $E_e(m) = c$. وهذا يعني أنه من غير الممكن أن نحدد d من خلال معرفتنا ب e، وهكذا يمكن أن ننشر e وتكون معلومة للجميع دون أن تعرف قيمة d ، وتكون E_e دالة اتجاه واحد ذات باب خلفي، الذي هو القيمة d .

يعتمد التشفير بالمفتاح العام على دالة اتجاه واحد ذات باب خلفي، فيكون e المفتاح العام ويكون d المفتاح الخاص الذي لا يعلمه إلا صاحب هذا المفتاح العام. وهكذا إذا ضمنا أن المفتاح e هو مفتاح عبيد حقيقة، فإن نظام التشفير بالمفتاح العام يضمن لزيد إقامة اتصال آمن مع عبيد هذا.

4.2.2 نظام RSA للتشفير بالمفتاح العام

في عام (1978) ظهر نظام RSA الذي سمي باسم مخترعيه الثلاثة وهم Rivest وShamirوAdleman. وخوارزمية RSA هي الآن الأكثر استعمالاً في كثير من التطبيقات، مثل: تطبيقات تأمين التجارة الإلكترونية، والرسائل البريدية. وقوة هذا النظام تعتمد على صعوبة تحليل الأعداد الكبيرة؛ فالمفاتيح تتولد من أعداد أولية بطول أكبر أو يساوي 100 رقم؛ ولهذا لكي نفهم RSA لا بد من شيء من مفاهيم نظرية الأعدد NumberTheory.

المفاهيم الأساسية لنظرية الأعداد:

سنسرد هذه المفاهيم بشكل نقاط:

الأعداد هي: $N = \{0, 1, 2, 3 \ldots.\}$ و $Z = \{0, 1, -1, \ldots.\}$

الأعداد الأولية: $Primes = \{2, 3, 5, 7 \ldots.\}$

لكل n تحليل وحيد إلى الأعداد الأولية. مثال: $60 = 2^3 * 3 * 5$

ضرب الأعداد سهل، وتحليلها إلى الأعداد الأولية صعب، إذ لا نستطيع تحليل أغلب الأعداد التي تتمثل في التمثيل الثنائي في أكثر من 1024بت

القواسم: نقول أن $a \neq 0$ تقسم b وتكتب $a|b$ إذا كان

$$\exists\, m, ma = b$$

$$\forall\, a, a \neq \mathbf{0} \Rightarrow a|\mathbf{0}$$

$$\forall\, b, ghmn, (b|g \wedge b|h) \Rightarrow b|(mg + nh)$$

لكل عددين أوليين مختلفين p و q : $(b|z \wedge q|z) \Rightarrow pq|z$

لكل $a, b \in N$ نعبر عن القاسم المشترك الأكبر بـ $gcd\ (a, b)$

مثال $\mathbf{60 = 2^3 * 3 * 5}$ و$\mathbf{14 = 2 * 7}$ فيكون $\boldsymbol{gcd}(\mathbf{60, 14}) = \mathbf{2}$

$a, b \in N$ هما نسبيًا أوليان إذا كان $\boldsymbol{gcd}(\boldsymbol{a}, \boldsymbol{b}) = \mathbf{1}$

يمكن حساب gcd سريعًا باستعمال خوارزمية اقليد

(Euclid's Algo)

```
Euclid(a, b)
1 if b = 0
2 then return a
3 else return Euclid(b, a mod b)
```

مثال:

$$\gcd(60,14) : 60 = 4 * 14 + 4$$
$$\gcd(14,4) : 14 = 3 * 4 + 2$$
$$\gcd(4,2) : 4 = 2 * 2$$

و تعمم الطريقة فنستطيع حساب لكل $x, y \in Z$ حيث يكون

$$\gcd(a, b) = xa + yb$$

```
Extended-Euclid(a, b)
1 if b = 0
2 then return (a, 1, 0)
3 (d', x', y') := Extended-Euclid(b, a mod b)
4 (d, x, y) := (d', y', x' – [a/b]y') ; [a/b] is the quotient of the
division (for a = qb + r).
5 return (d, x, y)
```

مثال :

$$2 = 14 - 3 * 4 = 14 - 3(60 - 4 * 14)$$
$$= -3 * 60 + 13 * 14$$

$\forall a, n. \exists q, r. a = q \times n + r where 0 \leq r < n$ وتكون r هنا باقي القسمة ونكتبها $a \ mod \ n = r$

$a, b \in Z$ يعتبران منسجمين في باقي القسمة على n إذا كان $a \ mod \ n = b \ mod \ n$ ونكتب خاصية الانسجام.

$a \equiv b \ (mod \ n)$ أو $a \equiv_n b$

$\equiv_n$ تمثل علاقة تكافؤ رياضية ونلاحظ أن

$$a \equiv_n b \Leftrightarrow a = qn + r \ and \ b$$
$$= q'n + r \ for \ some \ q,$$
$$q' \Leftrightarrow a - b = (q' - q)n \ \Leftrightarrow n|(a - b)$$

باقي القسمة الحسابي $\boldsymbol{mod}$ يملك الخاصيات التالية:

$$a \ \Delta \ b \equiv_n (amod \ \mathrm{n}) \ \Delta \ (bmodn), \Delta \in \{+, -, *\}$$

يعني:

$$a \ \Delta \ bmodn = [(amod \ \mathrm{n}) \ \Delta \ (bmodn)] \ modn$$

إذا كانت $a * b \equiv_n a * c$ و a نسبيا أولي لـ n إذن $b \equiv_n c$

مثال1:

$$\mathbf{2} = (\mathbf{5} \times \mathbf{6})\boldsymbol{mod} \ \mathbf{4}$$
$$= [(\mathbf{5} \ \boldsymbol{mod} \ \mathbf{4}) \times (\mathbf{6} \ \boldsymbol{mod} \ \mathbf{4})] \ \boldsymbol{mod} \ \mathbf{4}$$
$$= (\mathbf{1} \times \mathbf{2})\boldsymbol{mod} \ \mathbf{4}$$

مثال2:

$$\mathbf{8 * 4 \equiv_3 8 * 1 \Rightarrow 4 \equiv_3 1}$$

قاعدة: لنفترض عددين أوليين نسبيّاً $a, b \in Z$. فإنه يوجد $c \in Z$ حيث يكون $\boldsymbol{bc \ mod \ a = 1}$ ويمكن لنا أن نحسب $\boldsymbol{b^{-1} mod \ a}$

قاعدة: Fermat الصغيرة: لكل عددين أوليين نسبيّاً a و n و n عدد أوليّ فهو $\boldsymbol{a^{n-1} \equiv_n 1}$

مثال:

$$4^6 mod\ 7 = 16 * 16 * 16\ mod\ 7 = 2 * 2 * 2\ mod\ 7 = 1$$

دالة Euler's Totient

$\varphi(n)$تمثل عدد الأعداد الموجبة التي هي أقل من وهي أولية نسبيّاً معn. أي:

$$\varphi(n)\ is\ the\ number\ of$$
$$a \in \{1, 2, 3, \dots, n-1\} with\ \ gcd(a, n) = 1$$
$$\varphi(1) = 1$$
$$\varphi(p) = p - 1$$

إن كانpعدداً أوليّاً، ويمكن أن نكتب تبعاً قاعدة Fermat الصغيرة كما يلي

$$a^{\varphi(n)} \equiv_n 1$$

لكل عددين أوليينpو q بحيث يكون $p \neq q$ و$n = p * q$ فإن

$$\varphi(n) = \varphi(p * q) = \varphi(p) * \varphi(q) = (p-1) * (q-1)$$

n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$\varphi(n)$	1	1	2	2	4	2	6	4	6	4	10	4	12	6	8

الطريقة السريعة لحساب أس العدد: لتكن

$g \in G, G = (H, o, e)$ حيث H مجموعة وo عملية تجميعية على H وe عنصر محايد في H وليكن $e = \sum_{i=0}^{k} e_i 2^i$ حيث e_i تأخذ قيمة 0 أو 1 فإنه يمكن حساب g^e بالمعادلة التالية:

$$g^e = g^{\sum_{i=0}^{k} e_i 2^i} = \prod_{i=0}^{k} (g^{2^i})^{e_i} = \prod_{0<i<k, e_i} g^{2^i}$$

من هذه المعادلة الأخيرة تولدت الفكرة التالية في حساب الأس وهي:

نحسب التربيعات المتتالية لـ g^{2^i} حيث $0 < i < k$

نحدد g^e كحاصل ضرب لهذه التربيعات g^{2^i} للقيم $e_i = 1$

كما يجدر ملاحظة أن $g^{2^{i+1}} = (g^{2^i})^2$

مثال: كيفية حساب $6^{73} mod\ 100$

- $73 = 1 + 2^3 + 2^6$
- $6^2 = 36;\ 6^{2^2} = 36^2 \equiv -4\ mod\ 100$
- $6^{2^3} = 16\ mod\ 100$
- $6^{2^4} \equiv 16^2 \equiv 56\ mod\ 100$
- $6^{2^5} \equiv 56^2 \equiv 36\ mod\ 100$
- $6^{2^6} \equiv -4\ mod\ 100$
- $6^{73} \equiv 6 * 6^{2^3} * 6^{2^6} \equiv 6 * 19 * (-4)\ mod\ 100 \equiv 16\ mod\ 100$

وبهذا تطلب عملية حساب هذا الأس 6 تربيعات و2 عملية ضرب، أي 8 عمليات ضرب، عوضًا عن 72 عملية ضرب. هناك طريقة أسرع في حساب الكلفة، وهي تحويل الأس إلى التمثيل الثنائي، ثم ننزع أول بت من جهة اليسار ونحسب لكل بت 0 عملية ضرب واحدة، وللبت 1 عمليتا ضرب. فمثلا التمثيل الثنائي ل73، هو1001001 إذن نحتسب 4*1+2*2=8 عمليات ضرب.

اختبار Fermat: يكلف اختبار ما إذا عدد ما موجب هو عدد أولي أو لا كلفة كبيرة، ولكن هنا طريقة وهي اختبار Fermat يمكننا من إثبات أن العدد أولي بنسبة احتمال عالية. هذا الاختبار يعمل كالآتي:

نختار عددًا موجبًا $a \in \{1, 2, \dots, n-1\}$

نحسب $y = a^{n-1}\ mod\ n$ باستعمال طريقة حساب الأس السريعة.

إذا كانت $y \neq 1$ فإن عدد n مركب وليس أوليًا.

إذا كانت $y = 1$ فلا ندري هل ما إذا كان عدد n مركب أو أولي، واحتمالية أن لا يكون أوليًا $\frac{1}{10^{13}}$

test: pick a < n : a is relatively prime to n
if (a^{n-1} mod n) != 1 then
n is not prime
else n is probable prime

مثال: لو كانت $11 * 31$ فإن $2^{340} \equiv_{341} 1$ فلا يمكن أن نستنتج شيئًا، ولكن لو حسبنا $3^{340} \equiv_{341} 56$ عرفنا أن n ليست عددًا أوليًا. وهناك تنبيه مهم جدا: وهو أن اختبار Fermat وإن أثبت أن n ليس عددًا أوليًا فإن ذلك لا يعني أنه وجد قاسمًا لـn، بل هو يثبت أن خاصية متوفرة في كل الأعداد الأولية غير متوفرة في هذا العدد فقط. ولهذا اختبار Fermat لا يمكن أن يستعمل في تحليل العدد إلى الأعداد الأولية.

4.2.3 توصيف خوارزمية RSA

تعمل هذه الخوارزمية بالشكل الآتي:

نقوم بتوليد عددين أوليين كبيرين مختلفين p و q

نحسب $n = pq$ و $\varphi(n) = (p-1) * (q-1)$

نختار $e, 1 < e < \varphi(n)$ ويكون أوليًا نسبيا لـ$\varphi(n)$

نحسب العدد الوحيد $d, 1 < d < \varphi(n)$ حيث يكون $ed \bmod \varphi(n) = 1$

نحصل على المفتاح العام (n, e)، والمفتاح الخاص (d, n)

للقيام بالتشفير نمثل الرسالة في شكل عدد $m \in \{0, \dots, n-1\}$، ثم نحسب الكتلة المشفرة $c = m^e \bmod n$

لفك التشفير نستعمل المفتاح الخاص d، ونحسب القيمة $m = c^d \bmod n$

يمكن إثبات صحة الخوارزمية بما يلي:

بما أن $ed \bmod \varphi(n) = 1$ فإنه يوجد $k \in K$ حيث .

الآن لِنَرَ كلتا الحالتين $gcd(m, p) = 1$ أولاً.

الحالة الأولى: إذا كان $gcd(m, p) = 1$ فإنه باستعمال قاعدة Fermat نحصل على المعادلة التالية: $m^{p-1} \equiv_p 1$ لنرفع الجميع إلى أس $k(q-1)$ ونضرب بـ m لنحصل على $m^{1+k(p-1)(q-1)} \equiv_p m$

الحالة الثانية: إذا كان $gcd(m,p) = p$ فإن آخر انسجام –بديهةً– صحيحٌ، لأن شِقي المعادلة منسجم على $\mathbf{0 \ mod \ p}$

وإذن في كلا الحالتين نحصل أن $m^{ed} \equiv_p m$ وبنفس طريقة التعليل $m^{ed} \equiv_q m$ لأن p و q عددان أوليان مختلفان، وينبني على هذا أن $m^{ed} \equiv_n m$، وبهذا نكون قد أثبتنَا $c^d \equiv_n (m^e)^d \equiv_n m$.

مثال لخوارزمية RSA:

لتكن $\mathbf{q = 71}$ و$\mathbf{p = 47}$ فتكون $\mathbf{n = pq = 3337}$

يجب أن يكون e أوليًّا نِسبيًّا لـ$\mathbf{3320 = 70 * 46 = \varphi(3337)}$

لنفترض أن اختيارًا عشوائيًّا لـe كان بالقيمة التالية $\mathbf{e = 79}$

لنحسب $\mathbf{d = 79^{-1} mod \ 3320 = 1019}$

ننشر المفتاح العام(n,e)، ونحتفظ بالمفتاح الخاص (d,n)

لتشفير رسالة m نقوم بتقطيعها إلى كتلات صغيرة مثلا تكون أقل من n

$\mathbf{m = 688 \ 232 \ 687 \ 966 \ 668}$

ونحسب $\mathbf{c_1 = 688^{79} mod \ 3337 = 1570}$ للكتلة الأولى، ثم الثانية إلى أن نأتي على كل الرسالة.

لفك التشفير نقوم بحساب $\mathbf{m_1 = 1570^{1019} mod \ 3337 = 688}$

قوة خوارزمية RSA ترجع إلى صعوبة تحليل الأعداد للأعداد الأولية، فلكي نعرف المفتاح الخاص لا بد من التعرف على $\boldsymbol{\varphi(n) = (p-1)(q-1)}$، وهذا لا يتم إلا بالتعرف على p و q فنرجع إلى ضرورة تحليل n للعوامل الأولية، الذي لا يعرف له اليوم أية خوارزمية تقوم بذلك في وقت polynomial، ولكن نظرًا للتقدم في تقنيات التحليل لا بد أن يكون طول n على الأقل 1024بت. وتظل الخوارزمية رهينة التقدم في علم الأعداد، إذ لو تمكنا من إيجاد خوارزمية فعالة للتحليل سيصبح RSA غير ذي أهمية.

5 مراجع إضافية

5.1 كتب

ننصح بالرجوع إلى الكتب التالية:

- Bruce Schneier. Applied Cryptography. John Wiley & Sons, New York, 1996.
- Dieter Gollmann. Computer Security. Wiley, 2000.
- Alfred J. Menezes, Paul C. van Oorschot, Scott A. Vanstone. Handbook of Applied Cryptography. CRC Press, 1996. Available online at http://cacr.math.uwaterloo.ca/hac/
- Arthur E. Hutt, Seymour Bosworth, Douglas B. Hoyt. Computer Security Handbook. John Wiley & Sons, 1995.
- Doug Stinson. Cryptography (Theory and Practice), CRC Press,2002.
- Simon Singh. The Code Book: The Science of Secrecy from Ancient Egypt to Quantum Cryptograhy. Anchor Books, 2000.
- David Kahn. The Codebreakers: The Story of Secret Writing. Scribner, 1996.
- Bruce Schneier. Secrets and Lies. Wiley, 2000.

5.2 مواقع

- Cryptography and ciphers:http://www.trincoll.edu/depts/cpsc/cryptography/index.html
- The Code Breakers: http://www.math.arizona.edu/~dsl/talk.htm
- The Enigma Machine: http://www.swimmer.org/morton/enigma.html,
- http://www.codesandciphers.org.uk and http://www.xat.nl/enigma
- Secret Code Breaker Online: http://codebreaker.dids.com/
- Beginners' Guide to Cryptography: http://www.ftech.net/~monark/crypto/index.htm
- Introduction to Cryptosystems:http://www.math.nmsu.edu/~crypto/Fundamentals.html
- Magic Decoder Game:http://raphael.math.uic.edu/~jeremy/crypt/cgi-bin/magic-gateway.cgi

- Making the Enigma ciphers for the film "Enigma":http://www.qufaro.demon.co.uk/enigmafilm/
- An online bibliography:http://www.ce.chalmers.se/~stefanp/Security/sec_bib.html
- The Cipher IEEE newsletter:http://www.ieee-security.org/cipher.html
- SANS Institute Reading Room:http://www.sans.org
- http://www.bakerstreet221b.de/canon/danc.htm

6 أهم مصطلحات الفصل

Transposition	التبادل
Substitution	التعويض
Code	الرمز
Cryptology	علم دراسة الكتابة السرية
Cipher	السيفر
Steganography	التعمية
Cryptography	التشفير
Cryptoanalysis	علم كسر الشيفرة
Encode	رمّز
Encrypt	شفّر
Encipher	سيفر
Security by obscurity	الأمن بالغموض
Information theory	نظرية المعلومات
Complexity theory	نظرية التعقيد
Block cipher	تشفير الكتل
Stream cipher	تشفير الدفق
Code cipher	التشفير بالترميز
Symmetric cryptography	التشفير التناظري
Asymmetric cryptography	التشفير غير التناظري
Repudiation	الإنكار
Cipher block	الكتلة المشفرة
Private key	المفتاح الخاص
Secret key	المفتاح السري
Shared key	المفتاح المشترك
Public key	المفتاح العام
Fiestel structure	بنية فيستل
Permutation	الإبدال
Completeness	خاصية الكمال
Avalanche effect	خاصية الانهيار البياني
Data encryption standard	التشفير المعياري للبيانات
Cipher Block Chaining	طريقة الربط بالكتل المشفرة

7 تمارين الفصل

1. افترض أن هناك ناديًا خاصًا صغيرًا يضم في عضويته 100 شخص، عندما يستخدم أعضاء هذا النادي التشفير التماثلي في مراسلاتهم أجب على الأسئلة التالية:

ج. كم عدد المفاتيح السرية التي يحتاجها أعضاء النادي لكي يتمكن جميع الأعضاء من التراسل بينهم؟

ح. كم عدد المفاتيح السرية التي يحتاجها أعضاء النادي في حال تمت الثقة في رئيس النادي من قبل كل عضو؟ بحيث عندما يريد عضو ما أن يرسل رسالة إلى عضو آخر فسوف يقوم بإرسالها أولاً إلى رئيس النادي، ومن ثم يقوم الرئيس بإعادة إرسالها إلى العضو الآخر الموجة له الرسالة.

خ. كم عدد المفاتيح السرية التي يحتاجها أعضاء النادي في حال قرر رئيس النادي أن يتوجب على العضوين المتراسلين أن يتصلا به أولاً، لكي يقوم هو بدورة بتكوين مفتاح سري مؤقت لإتمام عملية التراسل بينهما، بحيث يتم إرسال المفتاح السري لهما مشفرًا من قبل رئيس النادي.

2. عثر عدد من علماء الآثار على وثيقة جديدة كتبت بلغة مجهولة، وبعد مدة من الزمن وجدوا في نفس المكان لوح يتضمن جملة بنفس اللغة مع ترجمة لها إلى اللغة اليونانية، وباستخدام ذلك اللوح تمكنوا من قراءة الوثيقة. ما التعدي أو الهجوم الذي استخدمه علماء الآثار في هذه الحالة؟

3. يستخدم عليّ أسلوب التشفير المضاعف عندما يريد أن يرسل رسالة سرية من جهازه الحاسوبي إلى أحد زملائه، اعتقادًا منه بأن تشفير الرسالة مرتين باستخدام مفتاح مختلف في كل مرة سوف يجعل من الرسالة المرسلة أكثر أمانًا. هل في تعتقد أن هذا الأسلوب صحيح؟ علل.

4. لدى علي رسالة طويلة يريد أن يرسلها بشكل سري وذلك بتشفيره باستخدام طريقة استبدال الحروف البدائية

MonoAlphabeticSubstitution، وهـو يعتقـد أنـه فـي حـال تـم ضـغط الرسالة أولاً فقد يعزز حماية الرسالة من التعدي عليها باستخدام ثغرة تكرار الحروف الفردية. في اعتقادك أنت: هل هذا صحيح وهل الأفضل ضغط الرسالة قبل التشفير أو بعده؟ ولماذا؟

5. قم بتشـفير الرسـالة التاليـة: "the house is being sold tonight"، وذلـك باستخدام إحدى الخوارزميات التالية، مع مراعاة إهمال الفراغات بين الأحـرف، ومـن ثـم قـم بفـك التشـفير للحصـول علـى الـنص الأصـلي للرسـالة:
   د. خوارزمية فيجينر "Vigenere" باستخدام كلمة المفتاح "dollars"
   ذ. خوارزمية المفتاح الآتي "Autokey" باستخدام المفتاح "7"
   ر. خوارزميـة اللعـب الحـر "Playfair" باسـتخدام المفتـاح المكـون مـن في نص الرسالة؟

6. يقـوم أحمـد بقـراءة كتـاب غـامض يتضـمن كتابـات مشـفرة فـي أحـد أجـزاء الكتـاب، إعطـاء المؤلـف نصًا مشـفرًا هـو "CIW" وبعـد فقـرتين أخبـر المؤلـف القـارئ بـأن أسـلوب التشـفير المسـتخدم هـو أسـلوب الإسـاحة وأن النص الأصلي هو كلمـة "yes". في الفصـل التالي وجـد بطـل القصـة لـوح فـي الكهـف محفـور عليـه الـنص المشـفر التـالي: "XVIEWYWI". عندها اكتشـف أحمد مباشرة المعنى الحقيقـي للنص المشـفر. مـا نـوع التعـدي أو الهجـوم الـذي باشـره أحمـد هنـا ومـا الـنص الأصلي؟

7. افترض أن مفتاح التشـفير في نظام تشـفير النقـل هـو (3,2,6,1,5,4)، أوجد مفتاح فك التشـفير؟

8. بين تمثيل مصفوفة مفتاح تشـفير خوارزميـة النقـل باسـتخدام المفتـاح التالي: (3,2,6,1,5,4)، ومن ثم أوجد تمثيل مصفوفة مفتاح فك تشفير خوارزمية النقل

9. في خوارزميـة تشـفير بوليبيـوس "Polybius"، افتـرض أن كـل حـرف تـم تشـفيره كـرقمين صـحيحين، ومفتـاح التشـفير عبـارة عـن مصـفوفة أحـرف مقـاس 5×5 كمـا هـو في خوارزميـة اللعـب الحـر. الـنص الأصلي مكـون مـن الأحـرف داخـل المصـفوفة، و الـنص المشـفر عبـارة عـن الـرقمين الصـحيحين (كـل واحـد بيـن الـرقم 1 و 5) الـذين يمثلـون أرقـام

أعمدة وصفوف المصفوفة. قم بتشفير النص التالي"Anexercise" باستخدام خوارزمية اللعب الحر والمفتاح التالي:

	1	2	3	4	5
1	z	q	p	f	E
2	y	r	o	g	D
3	x	s	n	h	C
4	w	t	m	i/j	B
5	v	u	l	k	A

10. تقوم شفرة الإبدال بـ:

ز. إخفاء الأحرف الحقيقية للرسالة، بالإضافة إلى إخفاء تواتر الأحرف وأنماطها في الرسالة.

س. إخفاء الأحرف الحقيقية للرسالة، ولكنها لا تخفي تواتر الأحرف وأنماطها في الرسالة.

ش. عدم إخفاء الأحرف الحقيقية للرسالة، بالإضافة إلى إخفاء تواتر الأحرف وأنماطها في الرسالة.

ص. إخفاء تواتر الأحرف وأنماطها في الرسالة، ولكنها لا تخفي الأحرف الحقيقية للرسالة.

11. الطول الفعلي لنظام التشفير DES هو:

أ. 16 بت. ب. 64 بت.

ت. 56 بت. ث. جميع ما ذكر.

12. من الأساليب التي يتبعها المهاجم للتغلب على التشفير، ومن ثم فك الشفرة:

ض. من خلال محاولة كسر المفتاح (Through brute force against the key).

ط. من خلال استغلال نقاط الضعف في خوارزمية التشفير (Through weaknesses in the algorithm).

ظ. من خلال استغلال نقاط الضعف في أنظمة المساعدة (Through weaknesses in the surrounding system)

ع. جميع ما ذكر.

13. اشرح بالرسم نظام التشفير 3DES المطور.

14. في نظام التشفير RSA افترض أن القيم التي يحتاجها أحمد لتوليد المفاتيح الخاصة به هي:

$$p = 17q = 7$$
$$e = 5d = 77$$

غ. أوجد المفتاح العام والمفتاح الخاص لأحمد.

ف. قم بتشفير الرسالة التالية: 2، 10 لإرسالها إلى أحمد.

15. ما شكل الرسالة بعد فك شفرتها من قبل المستقبل.

16. في الخطوة النهائية من تطبيق الخوارزمية الإقليدية(Euclid'salgorithm) لإيجاد $gcd(m,n)$ ، ، نحصل على قيم كل من u و v بحيث أن$um + vn = 0$، هل $luml$ (والتي تساوي $lvnl$) هي المضاعف المشترك الأصغر لكل من m و n ؟

17. هل من المحتمل أن تكون قيمة$\varphi(n)$ أكبر من n ؟

18. إذا كانت قيمة $n = p_1^{\alpha 1} p_2^{\alpha 2} p_3^{\alpha 3} \dots p\, p_k^{\alpha k}$، حيث أن$p_i$ عددا أولي، ما قيمة $\varphi(n)$.

19. في خوارزمية RSA للتشفير، هل من الممكن لأكثر من d لتعمل مع المعطىe ، p و q ؟

20. في خوارزمية RSA للتشفير، عند معرفة أن قيم كل من الاعداد الأولية p و q متساوية في الحجم تقريبًا، بشكل تقريبي كم حجم $\varphi(n)$ مقارنة بـ n ؟

21. اذكر الفرق بين الأعداد الأولية والأعداد المركبة.

22. ما المقصود بالعدد الأولي النسبي.

23. عرف نظرية Fermat الصغيرة وبين تطبيقاتها.

24. عرف نظرية Euler وبَيِّنْ تطبيقاتها.

25. عرف الخوارزميات المتقطعة، واشرح أهميتها في حل المعادلات الخوارزمية.

26. أوجد قيمة كل من $\varphi(101)$، $\varphi(100)$، $\varphi(80)$، $\varphi(32)$، $\varphi(29)$.

27. بين أن $2^{24}-1$ و $2^{16}-1$ أعداد مركبة، يمكنك استخدام العبارة $(a^2 - b^2)$.

28. أوجد الناتج لكل من التالي باستخدام نظرية Fermat الصغيرة

أ. $13\ mod\ 5^{15}$

ب. $17\ mod\ 15^{18}$

ت. $17\ mod\ 456^{17}$

ث. $101\ mod\ 145^{102}$

29. أوجد الناتج لكل من التالي باستخدام نظرية Fermat الصغيرة

ج. $13\ mod\ 5^{-1}$

ح. $17\ mod\ 15^{-1}$

خ. $41\ mod\ 27^{-1}$

د. $101\ mod\ 70^{-1}$

30. أوجد الناتج لكل من التالي باستخدام نظرية Euler

ذ. $77\ mod\ 5^{-1}$

ر. $323\ mod\ 16^{-1}$

ز. $403\ mod\ 20^{-1}$

س. $667\ mod\ 44^{-1}$

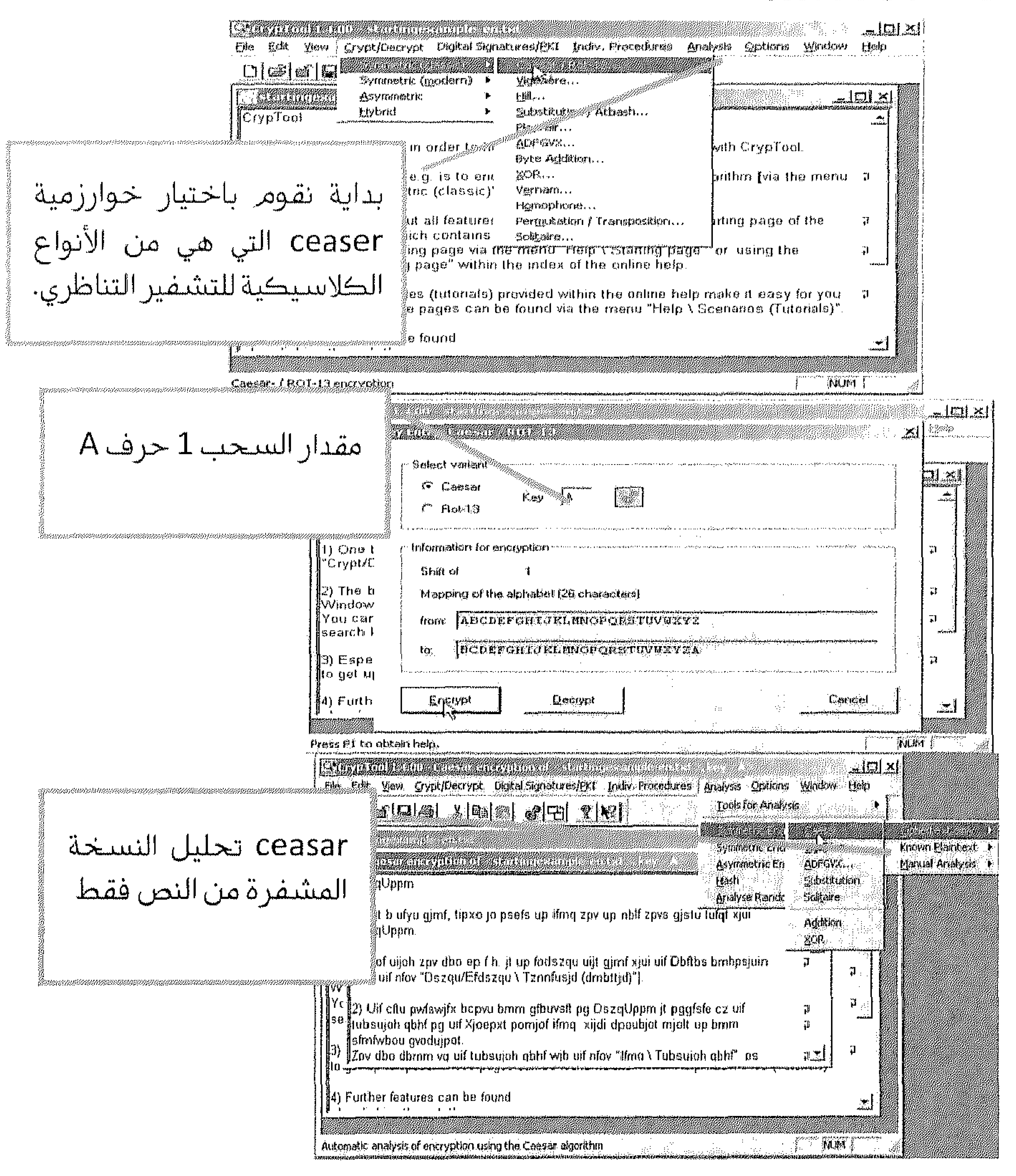
31. حدد أي من الأعداد الصحيحة التالية تنجح في تحقيق اختبار فيرمات الأولي باستخدام الأساس 2:

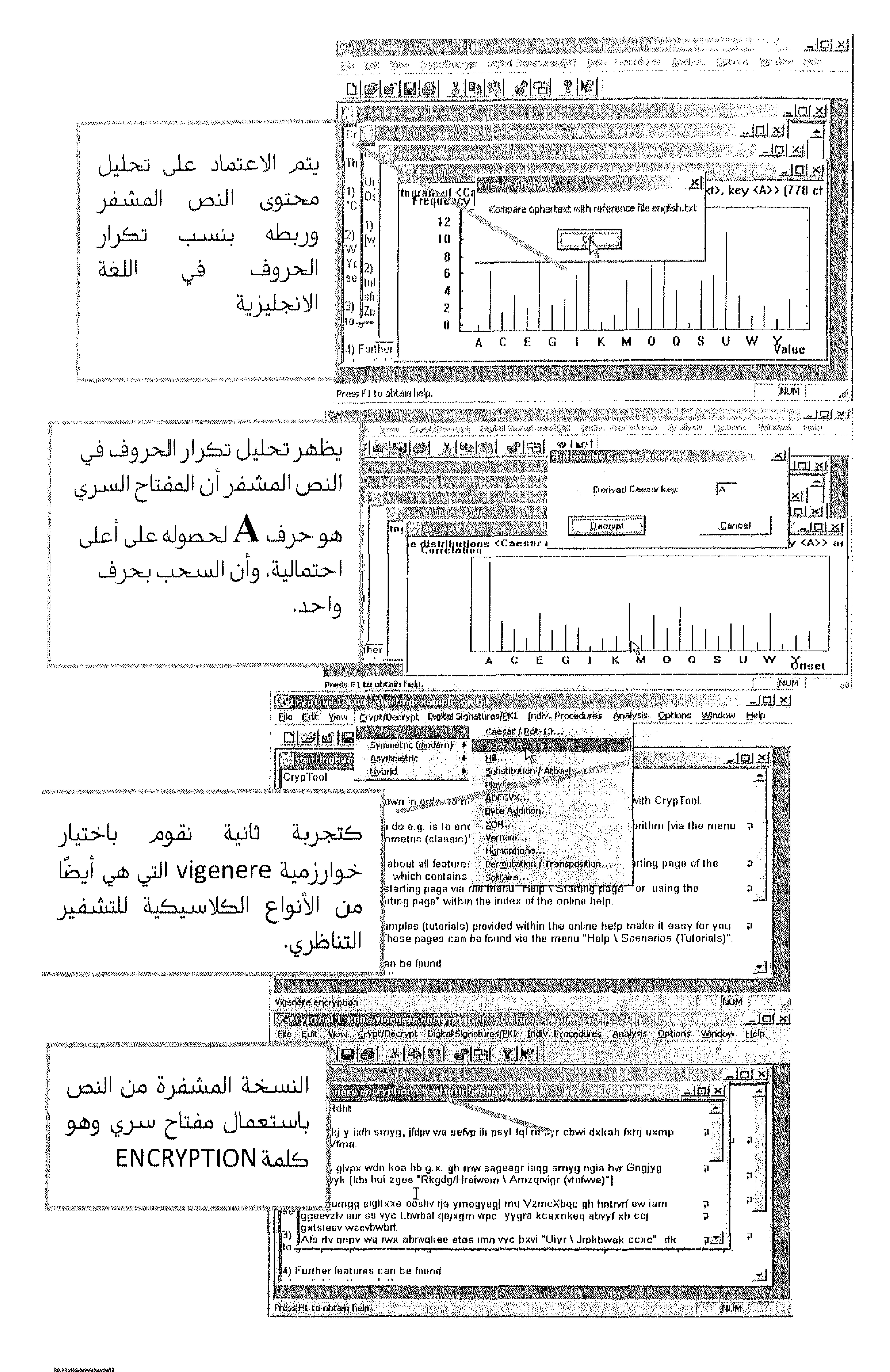
$100, 110, 130, 150, 200, 250, 271, 341, 561$

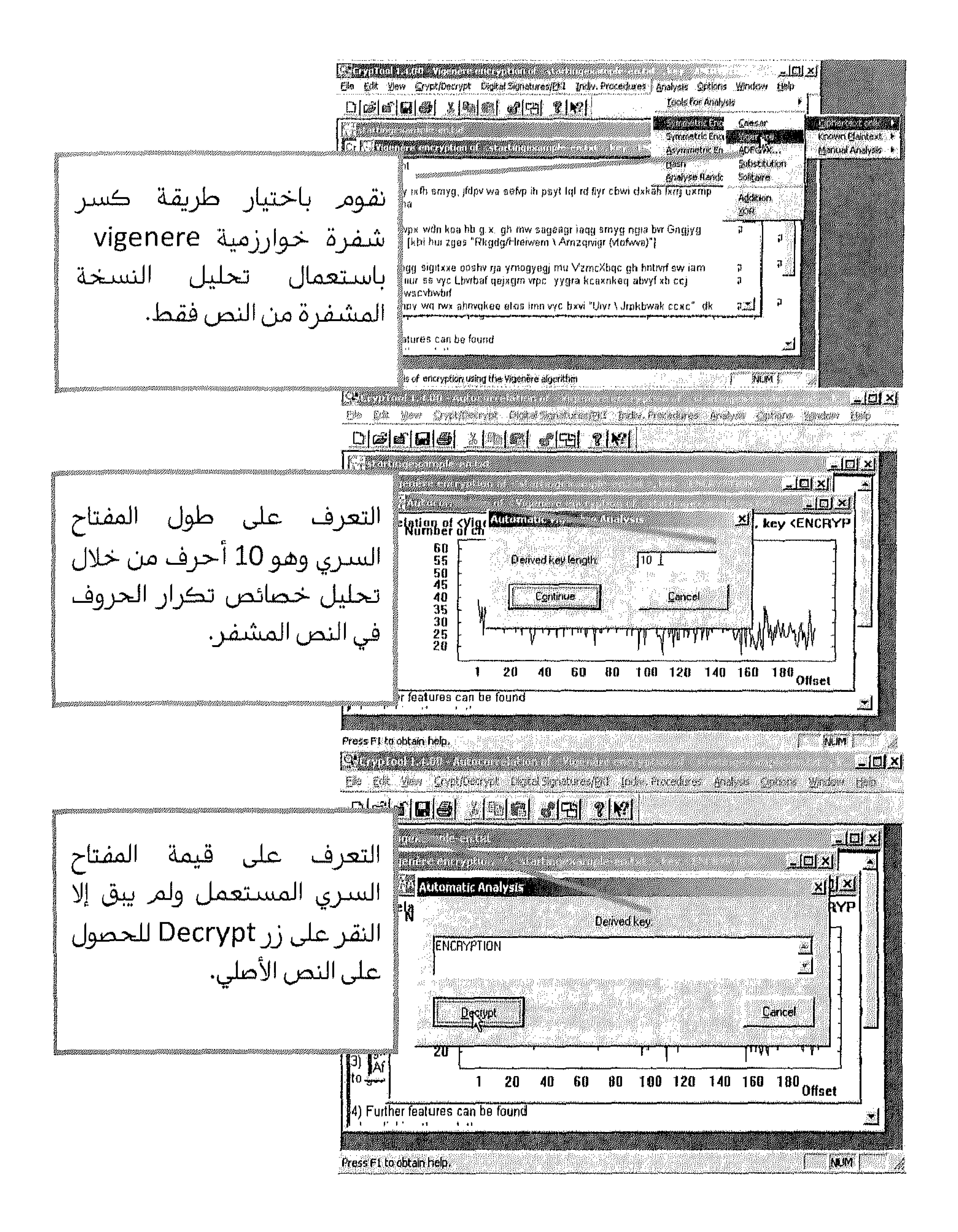
32. اكتب خوارزمية بأسلوب سيدوكود pseudocode اختبار Fermat الأولي.

8 ملحق رقم 1

كسر النصوص المشفرة عن طريق تحليل تردد الحروف باستعمال أداة CRYPTOOL




يتم الاعتماد على تحليل محتوى النص المشفر وربطه بنسب تكرار الحروف في اللغة الانجليزية
يظهر تحليل تكرار الحروف في النص المشفر أن المفتاح السري هو حرف A لحصوله على أعلى احتمالية، وأن السحب بحرف واحد.
كتجربة ثانية نقوم باختيار خوارزمية vigenere التي هي أيضًا من الأنواع الكلاسيكية للتشفير التناظري.
النسخة المشفرة من النص باستعمال مفتاح سري وهو كلمة ENCRYPTION



نقوم باختيار طريقة كسر شفرة خوارزمية vigenere باستعمال تحليل النسخة المشفرة من النص فقط.
التعرف على طول المفتاح السري وهو 10 أحرف من خلال تحليل خصائص تكرار الحروف في النص المشفر.
التعرف على قيمة المفتاح السري المستعمل ولم يبق إلا النقر على زر Decrypt للحصول على النص الأصلي.
Derived key length:
10
Continue
Cancel
Press F1 to obtain help.
Automatic Analysis
Derived key
ENCRYPTION
Decrypt
Cancel
4) Further features can be found
Offset
Press F1 to obtain help.

9 ملحق رقم 2

مثال تطبيقي على خوارزمية DES:

ليكن المفتاح المعطى ممثلا في نظام Hexa Decimal:

DE,10,9C,58,E8,A4,A6,30

فيكون تمثيله في النظام الثنائي على شكل مصفوفة كما يلي:

1	1	0	1	1	1	1	0	Bits 1-8
0	0	0	1	0	0	0	0	Bits 9-16
1	0	0	1	1	1	0	0	Bits 17-24
0	1	0	1	1	0	0	0	Bits 25-32
1	1	1	0	1	0	0	0	Bits 33-40
1	0	1	0	0	1	0	0	Bits 41-48
1	0	1	0	0	1	1	0	Bits 49-56
0	0	1	1	0	0	0	0	Bits 57-64

ثم يقع نزع البتات الثمانية ليصبح المفتاح:

1	1	0	1	1	1	1
0	0	0	1	0	0	0
1	0	0	1	1	1	0
0	1	0	1	1	0	0
1	1	1	0	1	0	0
1	0	1	0	0	1	0
1	0	1	0	0	1	1
0	0	1	1	0	0	0

ثم نقوم بعملية الإبدال PC-1 فنحصل على:

0	1	1	1	0	1	0
1	0	0	0	1	1	0
0	1	1	0	0	0	1
0	0	0	1	0	0	0
0	1	0	0	0	0	0
1	0	1	1	0	0	1
0	1	0	0	0	1	1
1	0	1	1	1	1	1

ونقسم المفتاح إلى نصفين أيمن و أيسر كما يلي:

C[0]								D[0]						
0	1	1	1	0	1	0		0	1	0	0	0	0	0
1	0	0	0	1	1	0		1	0	1	1	0	0	1
0	1	1	0	0	0	1		0	1	0	0	0	1	1
0	0	0	1	0	0	0		1	0	1	1	1	1	1

نقوم بتوليد المفاتيح بالسحب مرة ببت ومرة ب2بت حسب رقم الجولة كما هو موضح بالشكل التالي:

Iteration #	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Left Shifts	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

لنحصل على 16 نصف أيمن و16 نصف أيسر للمفاتيح التي ستستعمل في جولات التشفير كما في الشكل التالي:

C[0]
0 1 1 1 0 1 0 1 0 0 0 1 1 0 0 1 1 0 0 0 1 0 0 0 1 0 0 0

D[0]
0 1 0 0 0 0 0 1 0 1 1 0 0 1 0 1 0 0 0 1 1 1 0 1 1 1 1 1

C[1]
1 1 1 0 1 0 1 0 0 0 1 1 0 0 1 1 0 0 0 1 0 0 0 1 0 0 0 0

D[1]
1 0 0 0 0 0 1 0 1 1 0 0 1 0 1 0 0 0 1 1 1 0 1 1 1 1 1 0

C[2]
1 1 0 1 0 1 0 0 0 1 1 0 0 1 1 0 0 0 1 0 0 0 1 0 0 0 0 1

D[2]
0 0 0 0 0 1 0 1 1 0 0 1 0 1 0 0 0 1 1 1 0 1 1 1 1 1 0 1

C[3]
0 1 0 1 0 0 0 1 1 0 0 1 1 0 0 0 1 0 0 0 1 0 0 0 0 1 1 1

D[3]
0 0 0 1 0 1 1 0 0 1 0 1 0 0 0 1 1 1 0 1 1 1 1 1 0 1 0 0

C[4]
0 1 0 0 0 1 1 0 0 1 1 0 0 0 1 0 0 0 1 0 0 0 0 1 1 1 0 1

D[4]
0 1 0 1 1 0 0 1 0 1 0 0 0 1 1 1 0 1 1 1 1 1 0 1 0 0 0 0

C[5]
0 0 0 1 1 0 0 1 1 0 0 0 1 0 0 0 1 0 0 0 0 1 1 1 0 1 0 1

D[5]
0 1 1 0 0 1 0 1 0 0 0 1 1 1 0 1 1 1 1 1 0 1 0 0 0 0 0 1

C[6]
0 1 1 0 0 1 1 0 0 0 1 0 0 0 1 0 0 0 0 1 1 1 0 1 0 1 0 0

D[6]
1 0 0 1 0 1 0 0 0 1 1 1 0 1 1 1 1 1 0 1 0 0 0 0 0 1 0 1

C[7]
1 0 0 1 1 0 0 0 1 0 0 0 1 0 0 0 0 1 1 1 0 1 0 1 0 0 0 1

D[7]
0 1 0 1 0 0 0 1 1 1 0 1 1 1 1 1 0 1 0 0 0 0 0 1 0 1 1 0

C[8]
0 1 1 0 0 0 1 0 0 0 1 0 0 0 0 1 1 1 0 1 0 1 0 0 0 1 1 0

D[8]
0 1 0 0 0 1 1 1 0 1 1 1 1 1 0 1 0 0 0 0 0 1 0 1 1 0 0 1

C[9]
1 1 0 0 0 1 0 0 0 1 0 0 0 0 1 1 1 0 1 0 1 0 0 0 1 1 0 0

D[9]
1 0 0 0 1 1 1 0 1 1 1 1 1 0 1 0 0 0 0 0 1 0 1 1 0 0 1 0

C[10]
0 0 0 1 0 0 0 1 0 0 0 0 1 1 1 0 1 0 1 0 0 0 1 1 0 0 1 1

D[10]
0 0 1 1 1 0 1 1 1 1 1 0 1 0 0 0 0 0 1 0 1 1 0 0 1 0 1 0

C[11]
0 1 0 0 0 1 0 0 0 0 1 1 1 0 1 0 1 0 0 0 1 1 0 0 1 1 0 0

D[11]
1 1 1 0 1 1 1 1 1 0 1 0 0 0 0 0 1 0 1 1 0 0 1 0 1 0 0 0

C[12]
0 0 0 1 0 0 0 0 1 1 1 0 1 0 1 0 0 0 1 1 0 0 1 1 0 0 0 1

D[12]
1 0 1 1 1 1 1 0 1 0 0 0 0 0 1 0 1 1 0 0 1 0 1 0 0 0 1 1

C[13]
0 1 0 0 0 0 1 1 1 0 1 0 1 0 0 0 1 1 0 0 1 1 0 0 0 1 0 0

D[13]
1 1 1 1 1 0 1 0 0 0 0 0 1 0 1 1 0 0 1 0 1 0 0 0 1 1 1 0

C[14]
0 0 0 0 1 1 1 0 1 0 1 0 0 0 1 1 0 0 1 1 0 0 0 1 0 0 0 1

D[14]
1 1 1 0 1 0 0 0 0 0 1 0 1 1 0 0 1 0 1 0 0 0 1 1 1 0 1 1

C[15]
0 0 1 1 1 0 1 0 1 0 0 0 1 1 0 0 1 1 0 0 0 1 0 0 0 1 0 0

D[15]
1 0 1 0 0 0 0 0 1 0 1 1 0 0 1 0 1 0 0 0 1 1 1 0 1 1 1 1

C[16]
0 1 1 1 0 1 0 1 0 0 0 1 1 0 0 1 1 0 0 0 1 0 0 0 1 0 0 0

D[16]
0 1 0 0 0 0 0 1 0 1 1 0 0 1 0 1 0 0 0 1 1 1 0 1 1 1 1 1

نؤلف بين شقي كل مفتاح فمثلا للمفتاح الأول نحصل على المصفوفة التالية:

C[0]D[0]

0	1	1	1	0	1	0	bits 1-7
1	0	0	0	1	1	0	bits 8-14
0	1	1	0	0	0	1	bits 15-21
0	0	0	1	0	0	0	bits 22-28
0	1	0	0	0	0	0	bits 29-35
1	0	1	1	0	0	1	bits 36-42
0	1	0	0	0	1	1	bits 43-49
1	0	1	1	1	1	1	bits 50-56

ثـم نقـوم بعمليـة الإبـدال PC-2 فنحصـل علـى المفتـاح الأول للجولـة الأولى:

K[0]

0	1	0	0	0	0
1	0	0	1	1	0
0	0	1	1	0	1
1	0	0	0	1	1
0	1	0	0	0	1
1	0	0	0	0	1
1	1	1	1	0	1
0	1	1	1	0	0

وهكذا نفعل ببقية الأنصاف لتوليد 16 مفتاح، لكل جولة مفتاح خـاص بها.

الآن لتكن الكتلة المراد تشفيرها الكتلة الموالية:

86,	01010110
233,	11101001
158,	10011110
172,	10101100
222,	11011110
95,	01011111
244,	11110100
177,	10110001

فيكون تمثيلها في شـكل مصفوفة كما يلي:

0	1	0	1	0	1	1	0	Bits 1-8
1	1	1	0	1	0	0	1	Bits 9-16
1	0	0	1	1	1	1	0	Bits 17-24
1	0	1	0	1	1	0	0	Bits25-32
1	1	0	1	1	1	1	0	Bits 33-40
0	1	0	1	1	1	1	1	Bits 41-48
1	1	1	1	0	1	0	0	Bits 49-56
1	0	1	1	0	0	0	1	Bits 57-64

بعد عملية الإبدال الأولية تصبح كالآتي:

0	1	1	1	0	0	1	1
1	1	1	1	0	1	0	1
0	1	1	1	1	1	0	1
1	0	1	0	0	0	1	0
1	1	0	1	1	1	1	0
1	1	0	0	1	0	1	0
0	0	1	1	1	1	1	0
0	0	1	1	0	1	0	1

تقسم إلى شقين أيمن وأيسر:

L[0]								R[0]							
0	1	1	1	0	0	1	1	1	1	0	1	1	1	1	0
1	1	1	1	0	1	0	1	1	1	0	0	1	0	1	0
0	1	1	1	1	1	0	1	0	0	1	1	1	1	1	0
1	0	1	0	0	0	1	0	0	0	1	1	0	1	0	1

لنأخذ الآن الشق الأيمن المعني بالتشفير:

R[0]								
1	1	0	1	1	1	1	0	bits 1-8
1	1	0	0	1	0	1	0	bits 9-16
0	0	1	1	1	1	1	0	bits 17-24
0	0	1	1	0	1	0	1	bits 25-32

نقوم بتوسيع الشق الأيمن بالطريقة التي وصفنا في الخوارزمية فتصبح بحجم 48 بت كالآتي:

1	1	0	1	1	1
1	1	1	1	0	1
0	1	1	0	0	1
0	1	0	1	0	0
0	0	0	1	1	1
1	1	1	1	0	0
0	0	0	1	1	0
1	0	1	0	1	1

ثم نقوم بعملية XOR مع مفتاح التشفير $k[0]$ لنحصل على المصفوفة التالية:

1	0	0	1	1	1
0	1	1	0	1	1
0	1	1	1	0	0
1	1	1	0	0	1
0	1	1	1	1	0
0	1	1	1	0	1
1	1	1	0	1	1
1	1	0	1	1	1

نقسم الناتج إلى 8 كتلات بحجم 6 بت لإدخالها في المصفوفات التعويضية:

B[1]
1 0 0 1 1 1
B[2]
0 1 1 0 1 1
B[3]
0 1 1 1 0 0
B[4]
1 1 1 0 0 1
B[5]
0 1 1 1 1 0
B[6]
0 1 1 1 0 1
B[7]
1 1 1 0 1 1
B[8]
1 1 0 1 1 1

نحدد رقم السطر m ورقم العمود n ثم نستخرج القيمة التعويضية من المصفوفة الأولى:

Substitution Box 1 (S[1])

14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

S[1][2][3]=2

B[1]

1	0	0	1	1	1	
1	2	3	4	5	6	bit order

m=11=3
n=0011=3

كذلك نفعل بباقي الكتلات مع المصفوفات التعويضية لاستخراج القيم التعويضية:

B[2]

0 1 1 0 1 1

m=01=1
n=1101=13

S[2]

15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9

S[2][1][13]=9

B[3]

0 1 1 1 0 0

m=00=0
n=1110=14

S[3]

10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8
13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1
13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7
1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12

S[3][0][14]=2

B[4]

1 1 1 0 0 1

m=11=3
n=1100=12

S[4]

7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15
13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9
10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4
3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14

S[4][3][12]=12

S[5]

2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6
4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3

S[5][0][15]=9

B[6]

0 1 1 1 0 1

m=01=1
n=1110=14

S[6]

12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11
10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8
9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6
4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13

S[6][1][14]=3

B[7]

1 1 1 0 1 1

m=11=3
n=1101=13

S[7]

4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1
13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6
1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2
6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12

S[7][3][13]=2

B[8]

1 1 0 1 1 1

m=11=3
n=1011=11

S[8]

13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7
1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
7	11	4	1	9	12	14	2	0	6	10	13	14	3	5	8
2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11

S[8][3][11]=0

لنقلص في النهاية 48 بت إلى 32 بت التالية:

B[1]=S[1][3][3]=2=0010
B[2]=S[2][1][13]=9=1001
B[3]=S[3][0][14]=2=0010

B[4]=S[4][3][12]=12=1100

B[5]=S[5][0][15]=9=1001

B[6]=S[6][1][14]=3=0011

B[7]=S[7][3][13]=2=0010

B[8]=S[8][3][11]=0=0000

B[1-8]

0	0	1	0	1	0	0	1	0	0	1	0	1	1	0	0	1	0	0	1	0	0	1	1	0	0	1	0	0	0	0	0
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32

لنقوم بعملية الإبدال على الناتج فنحصل على:

0	0	1	0
0	0	0	1
0	0	1	0
1	0	0	0
0	1	1	1
0	1	1	0
0	1	0	0
0	1	0	0

ثم نقوم بعملية XOR بين الناتج وبين:

L[0]

0	1	1	1		0	0	1	0
0	0	1	1		0	0	0	1
1	1	1	1		0	0	1	0
0	1	0	1		1	0	0	0
0	1	1	1		0	1	1	1
1	1	0	1	XOR	0	1	1	0
1	0	1	0		0	1	0	0
0	0	1	0		0	1	0	0

لنحصل على الشق الأيمن للجولة الموالية وهو $R[1]$:

R[1]

0	1	0	1
0	0	1	0
1	1	0	1
1	1	0	1
0	0	0	0
1	0	1	1
1	1	1	0
0	1	0	0

أما $L[1]$ فيأخذ قيمة $R[0]$ وهكذا أنهينا الجولة الأولى، لنبدأ في الجولة الثانية ثم الثالثة إلخ.

الفَصْلُ الثَّالِث

أَمْنُ الشَّبَكَاتِ وَالْبْرُوتُوكُولات

يَهْدِفُ هَذَا الْفَصْلُ إِلَى:

1. التعريف بالمفاهيم الأساسية لأمن الشبكات.
2. تقديم أهم خدمات أمن الشبكات.
3. تقديم أهم نظم أمن الشبكات.

1 مقدمة الفصل

يقول وايتفيلد ديفي في رسالته »العشر سنوات الأولى للتشفير بالمفتاح العام« عام (1988): "وُلِدَ التشفيرُ في مايو عام (1975) ابنًا لمشكلتين: مشكلة توزيع المفاتيح، ومشكلة التوقيع الالكتروني. الاكتشاف لم يكن فقط حلاً ولكن لأن المشكلتين بدتا كما لو أنه لا حل لكل واحدة منهما يوجد مطلقًا، فكان أنْ جَاءَ الحل لكلتيهما في حزمة واحدة". ولكنه يقول في نفس الرسالة: "ما الفائدة من تطوير نظام تشفير غير قابل للاختراق إذا كان مستخدمو النظام مضطرين لنشر مفاتيحهم لمركز توزيع مفاتيح قابل للاختراق والسطو".

نتطرق في هذا الفصل لمشكلة توزيع المفاتيح، والتأكد من هوية الرسالة والمرسل عن طريق رمز التأكد من هوية الرسالة، وتقنية التوقيع الالكتروني.

2 تبادل المفاتيح وسرية البيانات

كل تقنيات التشفير توفر هذه الخدمة، فبالإمكان استعمال أي تقنية لتشفير البيانات إما بالتشفير التماثلي، أو بالمفتاح العام. ومع أن التشفير التناظري أسرع وأقل كلفة من التشفير بالمفتاح العام، إلا أنه سبق أن أشرنا لمشكلة التشفير التناظري، وهي تكمن في إدارة المفاتيح كما أشار إلى ذلك ديفي في الفقرة السابقة، إذ لو أن عندنا ألف مستخدم فإن التشفير التناظري يتطلب منا إدارة 499500 مفتاح مقارنة بـ2000 مفتاح فقط في التشفير بالمفتاح العام. فمع ظهور التشفير بالمفتاح العام، فإنه صار بالإمكان تلافي هذه المشكلة، وذلك إما بالاستعاضة عنه بنظام المفتاح العام، وإما باستعمال هذا النظام في توزيع مفاتيح التشفير التناظري، ومن ثَمَّ استعمال النظام التناظري في تشفير الرسائل، لأنه أسرع وأقل كلفة من التشفير بالمفتاح العام. فلتشفير الرسالة m يقع اختيار رقم عشوائي k الذي نستعمله في التشفير التناظري للرسالة $c_2 = E_k(m)$ ثم يقع تشفير هذا المفتاح بالمفتاح العام $c_1 = k^e mod\, n$ ويتم إرسال الزوج $(c_1 = k^e mod\, n, c_2 = E_k(m))$ للطرف المقابل على الشبكة، فيقوم بفك تشفير المفتاح بمفتاحه الخاص $k = {c_1}^d mod\, n$ ليستعمله بعد ذلك في فك تشفير الرسالة $m = D_k(c_2)$.

2.1.1 تبادل المفاتيح بطريقة DH (Diffie-Hellman)

تعتمد هذه الطريقة على صعوبة حساب اللوغاريثمات المحددة (Discrete Logs) لعدد ما.

تعريف اللوغاريثم المحدد:

نسمي s الجذر البدائي لعدد أولي p العدد الذي أسسه تولد $\mathbf{1, \ldots, p-1}$ فيكون $\boldsymbol{s \bmod p, s^2 \bmod p, \ldots, s^{p-1} \bmod p}$ مختلفة فيكون عندنا $\boldsymbol{\forall b \in Z. \exists i \in \{0, \ldots, p-1\}. b = s^i \bmod p}$. فإذا كان $b \in Z$ معطى لدينا فإن i هو ما نعبر عنه باللوغاريثم المحدد ل b للقاعدة $s, modp$

توصيف Diffie-Helmann:

- يشترك طرفا المخاطبة زيد وعبيد في s بدائي وجذر p، وهذان العددان يمكن معرفتها من الجميع أي قيم عامة.
- كل من زيد وعبيد يختار عددًا عشوائيًا أصغر من p (ليكن a لزيد وb لعبيد)
- يحسب زيد القيمة التالية $\alpha = s^a \bmod p$ وعبيد القيمة التالية $\beta = s^b \bmod p$
- يتبادل زيد وعبيد هذه القيم.
- يحسب زيد القيمة التالية.

$$k_a = \beta^a \bmod p = (s^b \bmod p)^a = (s^b)^a \bmod p$$

- يحسب عبيد القيمة التالية.

$$k_b = \alpha^b \bmod p = (s^a \bmod p)^b = (s^a)^b \bmod p$$

- نلاحظ أن $k_a = k_b$ وهذا هو المفتاح المشترك الذي سيكون بين زيد وعبيد.

نلاحظ أن أكبر نقطة قوة في DH هي تبادل مفتاح سري بدون اشتراط امتلاك أي معلومة مسبقة لدى المتخاطبين. ولكن هناك نقطتا ضعف، الأولى: أن هذا المفتاح المشترك غير متأكد من هويته. والثانية: أن المفتاح يمكن أن يتعرض للهجوم الموالي والذي يخل بسلامته.

- لتكن $\boldsymbol{p = rq + 1}$ إذن $\boldsymbol{s^q = s^{(p-1)/r}}$ تكون ذا ترتيب r أي $\boldsymbol{(s^{(p-1)/r})^r = 1}$

- في المرحلة الأولى يعوض الدخيل s^a و s^b بقيمتي s^{aq} وs^{bq} ويكون المفتاح المشترك بين زيد وعبيد والدخيل s^{abq} والتي لا تأخذ إلا على قيم r.
- الحل لهذه المشكلة أن تشفر قيم الأس المستعملة، ولكن هذا يتطلب مفتاحًا مشتركًا أيضًا.

2.1.2 طريقة الجمل ElGamal في التشفير باستعمال DH

طَوَّرَ هذه الطريقةَ المصريُّ طاهرُ الجمل عام (1985) حين استعمل طريقة DH مع إضافة دالة تشفير بالمفتاح التماثلي f (الشيء الذي يفترق عن DH مكتوب بالأحمر).

يشترك طرفا المخاطبة زيد وعبيد في s بدائي جذر و p وهذان العددان يمكن معرفتها من الجميع أي عامة.

- كل من زيد وعبيد يختار عددًا عشوائيًا أصغر من p (ليكن a لزيد وb لعبيد)
- يحسب عبيد القيمة التالية $\beta = s^b \ mod \ p$ ويرسلها لزيد.
- يحسب زيد القيمة التالية $\alpha = s^a \ mod \ p$ و $k = \beta^a \ mod \ p$ ويرسل لعبيد α مع $f_k(m)$
- يحسب عبيد $k = s^{ab} \ mod \ p$ ويستعمل k لفك شفرة $f_k(m)$

كما طور طاهر الجمل في ذات الورقة العلمية عام (1985) مع طريقة التشفير الآنفة الذكر طريقة توقيع إلكتروني اعتمدت بعد ذلك من معهد التقييس والتكنولوجيا الأمريكي NIST، وهي الآن تستعمل في أغلب البرامج والبروتوكولات المشهورة مثل نظام PGP للرسائل البريدية.

2.1.3 طريقة Massey-Omura

هذه الطريقة تعتمد على التشفير بدون استعمال المفاتيح المشتركة وتعتمد على صعوبة إيجاد اللوغاريثمات المحددة.

يملك طرفا الاتصال زيد وعبيد عددا أوليا p يكون معلوما للجميع

يختار زيد وعبيد عددين $e,d \in Z$ حيث $\boldsymbol{ed \bmod (p-1) = 1}$ وإذن يوجد عدد k بحيث $\boldsymbol{ed = k(p-1)+1}$

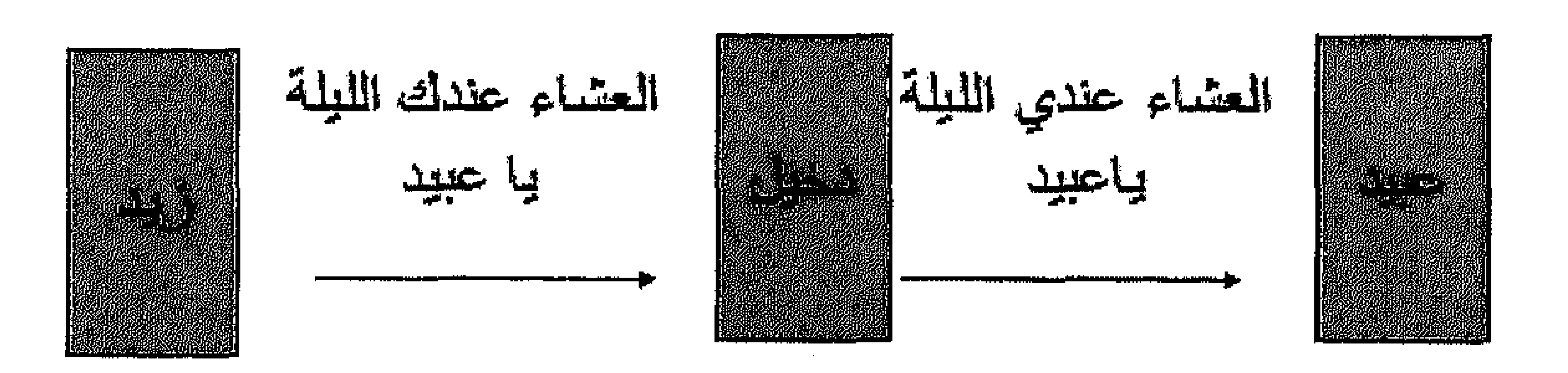
باستعمال قاعدة Euler يكون لكل $\boldsymbol{m \in \{1,\dots,p-1\}}$، $m^{ed} mod\, p =$

$$m^{k(p-1)} m \bmod p = m \bmod p = m$$

3 خدمة سلامة البيانات والتأكد من الهوية

تعتبر البيانات سليمة إذا لم يقع تغييرها بشكل غير مسموح به عند إنشائها وتبادلها على الشبكة، أو تخزينها عند مصدر موثوق. وإذا وقع أي تغيير في الرسالة من قبل دخيل ما، فإن هذا يعتبر مساسًا بخدمة سلامة البيانات (انظر إلى الشكل 1.3).

شكل 1.3 مثال إخلال بالسلامة



في نظم التشغيل يمكن استعمال طرق مراقبة صلاحيات الوصول للمعلومة، وفي الشبكات نستعمل تقنيات التشفير. أساسًا نستعمل الدوال ذات الاتجاه الواحد التي عرفناها رياضيًا في فصل مقدمة في علم التشفير، ونعبر عنها هنا بالبصمة. ونعتبر هذه الدالة آمنة في التشفير إذا توفرت فيها الخصائص الموالية:

1. من السهل حساب بصمة رسالة ما.

2. لو غيرنا تغييرًا طفيفًا في الرسالة، فالبصمة المولدة للرسالة المغيرة تختلف تمامًا عن بصمة الرسالة قبل التغيير.
3. من غير الممكن حسابيًا أن نحصل على الرسالة من خلال بصمتها.
4. من غير الممكن حسابيًا أن نحصل على رسالة أخرى لها نفس بصمة معينة عندنا.
5. من غير الممكن حسابيًا أن نحصل على رسالتين لهما نفس البصمة مطلقًا.

تعتبر الخاصيتان الرابعة والخامسة من أهم خصائص الخوارزمية، إذ بها تقاس قدرتها على الصمود للهجمات. عادة ما تدرس هاتان الخاصيتان تحت مسألة رياضية تسمى مفارقة عيد الميلاد (Birthday paradox) ونص المسألة هو: "كم من شخص يجب أن يكون في غرفة ما حيث تكون احتمالية أن أحدهم يحمل نفس تاريخ ميلادك أكثر من 0.5 $(p > 0.5)$؟

الجواب: هو إن كان عندنا n شخص فإن الاحتمالية هي $n/\mathbf{365}$ وإذن لو كانت $\boldsymbol{n} = \mathbf{183}$ فإن $(p > 0.5)$. والسؤال الثاني: كم من شخص يجب أن يكون في غرفة ما حيث تكون احتمالية أن اثنين منهم يحملان نفس تاريخ الميلاد أكثر من 0.5 $(p > 0.5)$؟

الجواب $\boldsymbol{n} = \mathbf{183}$. بشكل عام لتكن $\boldsymbol{h}$ لها $\mathbf{2}^{m}$ مخرجات محتملة. يجب أن تطبق $\boldsymbol{h}$ على $\mathbf{2}^{m/2}$ من المدخلات، وإذن تكون احتمالية الاشتباه أكبر من 0.5 $(p > 0.5)$. عندما نتحدث في الفقرات المقبلة على التوقيع الالكتروني فسنذكر هجوم عيد يوم الميلاد الذي يركز على تشابه التوقيعات الإلكترونية، وكيفية الحماية منه.

3.1 خوارزميات توليد البصمة

من أول الخوارزميات التي ظهرت في توليد البصمة خوارزمية رابن (Rabin) سنة (1978). وهي تقوم على تقسيم الرسالة M إلى كتل بحجم

محدد $b_1, \ldots, b_n$ ثم نقوم باستعمال التشفير التناظري، مثلا DES لحساب القيم التالية:

$h_0 = IV(initial\ value)\ and\ h_1 = E_{b_i}(h_{i-1})$ فهي بهذا تكون شبيهة تشفير النصوص الطويلة عن طريق الربط بالكتل المشفرة.

أما بالنسبة للطرق الحديثة، فمن أشهر خوارزميات توليد البصمات للرسائل مجموعة رونالد ريفست (MD family of Ron Rivest) وكذلك خوارزمية SHA-1 المعتمدة من طرف NIST في أمريكا كخوارزمية معيارية. كل الخوارزميات تولد بصمة ذات طول محدد بغض النظر عن طول الرسالة وقصرها. فمثلا خوارزميات MD تولد بصمة بطول 64بت، أما خوارزمية SHA-1 فتولد بصمة بطول 160 بت. تعمل جل هذه الخوارزميات في شكل جولات كما هو الحال في التشفير التماثلي. نعرض فيما يلي لتفصيل خوارزمية MD2:

3.1.1 خوارزمية MD2

هي دالة ذات اتجاه واحد طورها Ron Rivest عام (1989). الخوارزمية طورت للحاسوب ذي المعالج الذي يعمل على 8 بت حينها. ومع أن خوارزميات مثل MD4 وMD5 وSHA-1 ظهرت للوجود، إلا أن MD2 مازالت قيد الاستعمال جزئيًا مع خزارزمية RSA في البنية التحتية للمفتاح العام لتوليد شهادة الوثوقية certificates.

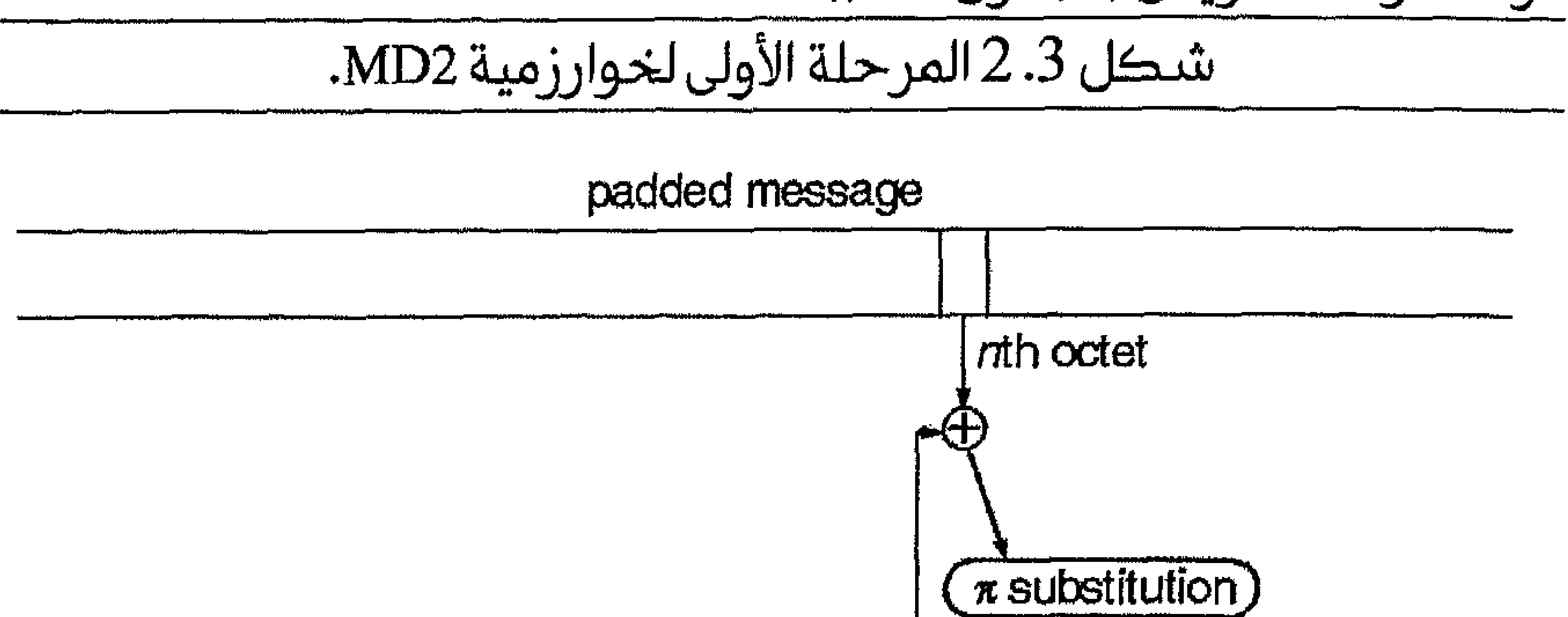
تولد MD2 بصمة بطول 128 بت أي 16 بايت. وتتولد كما يلي:

تكمل الرسالة حتى يصبح حجمها من مكررات 128 بت، ثم نضيف 128 بت كبصمة أولية للرسالة لنمرر الناتج بعد ذلك للعمليات الأساسية في الخوارزمية. يقع إكمال الرسالة بإضافة البايتات الناقصة بحيث تكون قيمة كل بايت منها العدد نفسه، فمثلا لنفترض الرسالة التالي: $abcdefghij$ فإن الناقص هو ست بايتات، فنضيف ست بايتات تحمل رقم 6 فتصبح الرسالة $\boldsymbol{abcdefghij666666}$.

تحسب البصمة الأولية التي طولها 16 بايت تكون بداية ذات قيمة صفر بأن نمر على كل الرسالة، ونأخذ في كل مرة بايت من الرسالة، وليكن البايت رقم n وبايت من البصمة برقم $(n-1)\ \boldsymbol{mod}\ 16$ ، ونقوم بعملية xor بينهما ثم عملية تعويض باستعمال مصفوفة التعويضπ (سميت بهذا

الاسم لأن الأعداد التي فيها مستنتجة من فاصلة عدد π)، ثم نقوم بعملية xor بين الناتج والبايت رقم $n \bmod 16$ في البصمة لنضع في النهاية القيمة الناتجة في بايت رقم $n \bmod 16$ في ذات البصمة (انظر إلى الشكل 3. 2 ومصفوفة التعويض (الجدول 3. 1)).

شكل 3. 2 المرحلة الأولى لخوارزمية MD2.



جدول 3. 1 مصفوفة التعويض π

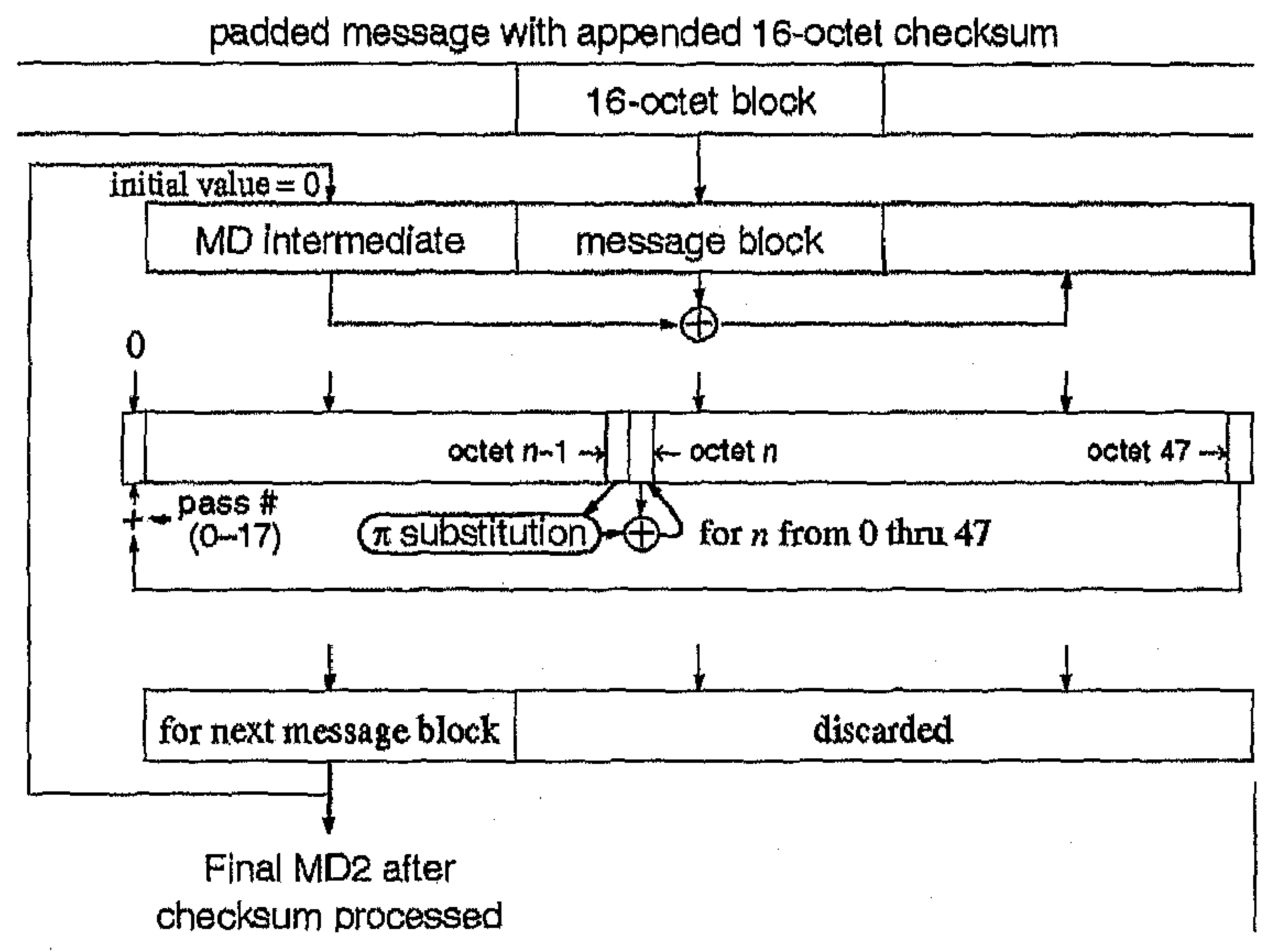
41	46	67	201	162	216	124	1	61	54	84	161	236	240	6	19
98	167	5	243	192	199	115	140	152	147	43	217	188	76	130	202
30	155	87	60	253	212	224	22	103	66	111	24	138	23	229	18
190	78	196	214	218	158	222	73	160	251	245	142	187	47	238	122
169	104	121	145	21	178	7	63	148	194	16	137	11	34	95	33
128	127	93	154	90	144	50	39	53	62	204	231	191	247	151	3
255	25	48	179	72	165	181	209	215	94	146	42	172	86	170	198
79	184	56	210	150	164	125	182	118	252	107	226	156	116	4	241
69	157	112	89	100	113	135	32	134	91	207	101	230	45	168	2
27	96	37	173	174	176	185	246	28	70	97	105	52	64	126	15
85	71	163	35	221	81	175	58	195	92	249	206	186	197	234	38
44	83	13	110	133	40	132	9	211	223	205	244	65	129	77	82

- نضيف هذه البصمة الأولية للرسالة، ثم نمر على كل الرسالة ونأخذ في كل مرة كتلة بحجم 16 بايت، ونقوم بعملية xor مع كتلة التي

ستحمل البصمة النهائية، ولكنها في البداية تكون مملوءة أصفاراً، فينتج عن ذلك كتلة ثالثة بحجم 16 بايت فنحصل عل كتلة بحجم 48 بايت (16 للبصمة الوسيطة + 16 للكتلة المأخوذة من الرسالة + 16 للناتج عن عملية xor).

- نقوم بعمل 18 جولة (0–17) في كل جولة نمر على كل البايتات، ونقوم بعملية تعويض باستعمال مصفوفة π للبايت السابق، ثم عملية xor بين الناتج والبايت الحالي، ونضع الناتج الأخير في البايت الحالي (البايت رقم 0 نحتسب معه رقم الجولة). وهكذا دواليك إلى أن تتم الجولة الأولى، لنبدأ جولة ثانية فثالثة إلى 18 جولة بين البايت الحالي والبايت السابق.
- نقوم بأخذ 16 البايت الأولى من الناتج لتكون هي الآن البصمة الوسيطة، ثم ننتقل لأخذ كتلة 16 بايت الموالية في الرسالة، ونقوم بنفس العمليات لهذه الكتلة الثانية، لنحصل على بصمة وسيطة جديدة تستعمل مع الكتلة الثالثة من الرسالة، وهكذا إلى أن نأتي عل كل الرسالة، وتكون البصمة النهائية آخر بصمة وسيطة نحصل عليها (انظر إلى الشكل 3.3)


شكل 3.3 خوارزمية MD2



3.1.2 أمثلة حسابية

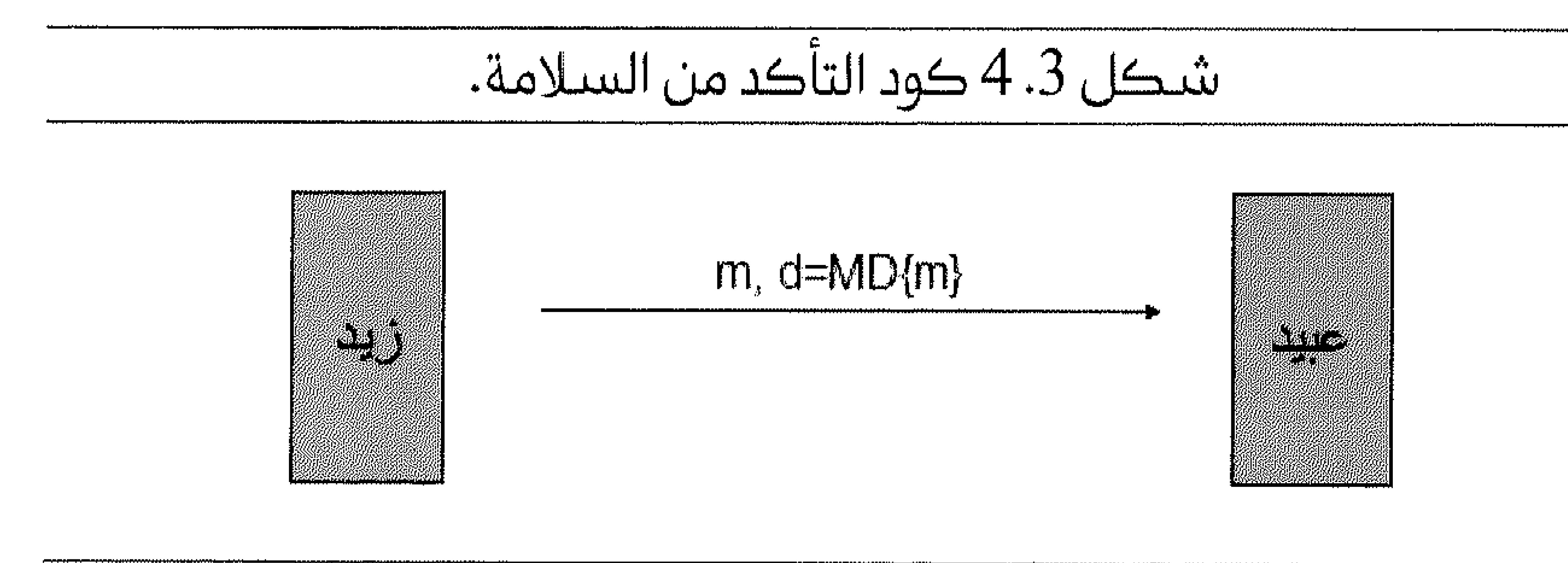
$$MD2("The\ quick\ brown\ fox\ jumps\ over\ the\ lazy\ dog") = 03d85a0d629d2442e987525319fc471$$

$$MD2("Thequickbrownfoxjumpsoverthelazycog") = 6b890c9292668cdbbfda00a4ebf31f05$$

نلاحظ أن الفرق بين الرسالتين في حرف واحد وهو حرف c ، عوضًا عن d في كلمة dog ومع هذا فإن البصمة اختلفت تمامًا مما يؤكد قوة الخوارزمية. طور رون ريفست فيما بعد خوارزمية MD3، ولكنها كسرت قبل أن تظهر للوجود كحال MD1 التي لم تنشر قط. ثم خوارزمية MD4 التي طورت لتعمل على المعالجات التي تعمل على 32 بت، وكان التركيز في تصميمها على السرعة. ثم خوارزمية MD5 التي كان الهدف من تطويرها لتكون أكثر أمنًا، وإن كان ذلك على حساب السرعة. ثم الآن MD6 التي دخلت في منافسة SHA-3، ولكن ريفست أعلن في يوليو (2009) سحبها من سباق المنافسة؛ لأنها لم تجهز بالشكل المطلوب إلى حينها.

3.2 **التأكد من سلامة الرسالة** (MIC)

شكل 3. 4 كود التأكد من السلامة.

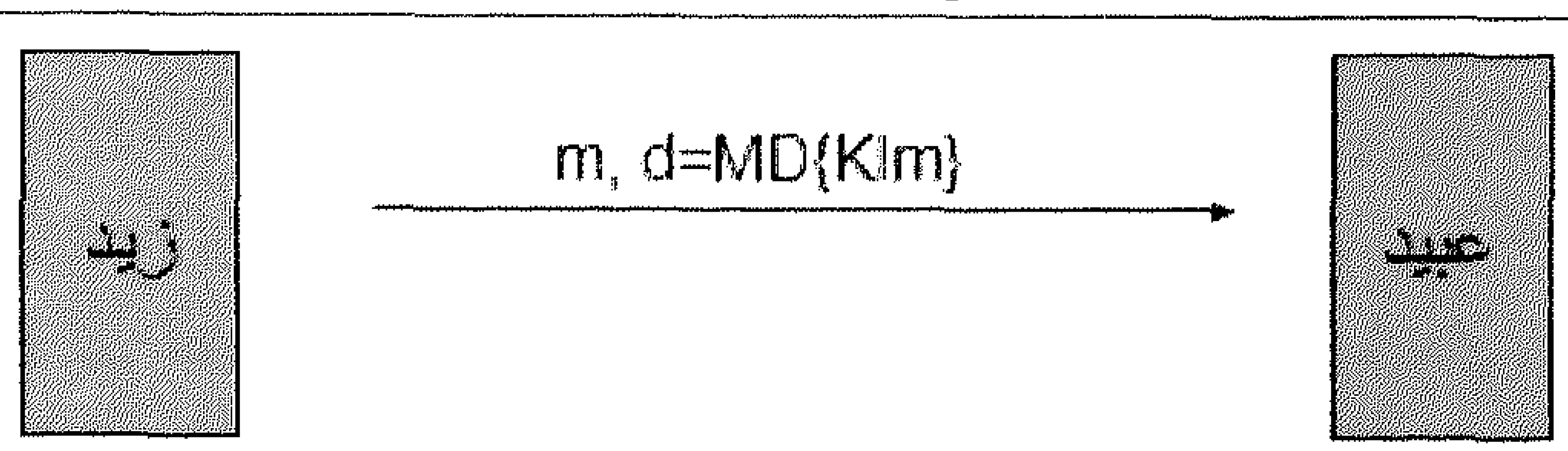


نقوم بالتأكد من سلامة الرسالة على الشبكة بأن نرسل الرسالة بشكلها النصي الواضح – أي غير مشفرة –، لأن الهدف هنا السلامة لا السرية، مع قيمة بصمتها المحسوبة عند زيد. لما تصل الرسالة عبيداً يقوم هو بدوره من جديد بحساب بصمة الرسالة التي وصلته، باستعمال نفس الخوارزمية التي استعملها

زيد، ثم يقارن هذه البصمة مع البصمة التي وصلته من زيد، فإن تساوتا فإن الرسالة وصلت سليمة، وإن افترقتا فهناك خلل في الرسالة.

3.3 التأكد من هوية الرسالة (MAC)

شكل 5.3 كود التأكد من هوية الرسالة.



يمكن استعمال خوارزميات توليد البصمة للتأكد من هوية الرسالة، وذلك بأن يرسل زيد الرسالة بشكلها الواضح، ويرسل معها البصمة. لحساب البصمة نقوم بإضافة قيمة المفتاح السري المشترك بين زيد وعبيد في بداية الرسالة، ونقوم بحساب البصمة على الرسالة مضمنة المفتاح السري. عندما تصل الرسالة لعبيد يقوم بحساب البصمة من جديد على الرسالة التي وصلته مضمنة المفتاح المشترك بينه وبين زيد بنفس طريقة التضمين، – أي بأن يجعله في بداية الرسالة –، ثم يقارن الناتج مع البصمة المرسلة من زيد. إن تساوت البصمتان فالرسالة جاءت من زيد، وإلا فلا تعتبر أنها جاءت من زيد. وهكذا نكون قد تأكدنا من هوية الرسالة.
بعض خوارزميات توليد البصمة عنده الخاصية التالية:

$$\begin{aligned} &if\ d = MD(x)\ then \\ &\quad for\ some\ y, d' = MD(x|y) = MD(x) + MD(y) = d + MD(y) \end{aligned}$$

فإنه يمكن للدخيل أن يقوم بتغيير رسالة زيد دون شعور عبيد بذلك، وذلك بأن يسحب الدخيل $< m, d >$ المرسلة من زيد ويعوضها ب$< m', d' >$ حيث تكون $m' = m|y$ و $d' = d + MD(y)$. يستقبل عبيد $< m', d' >$ ويقوم بحساب

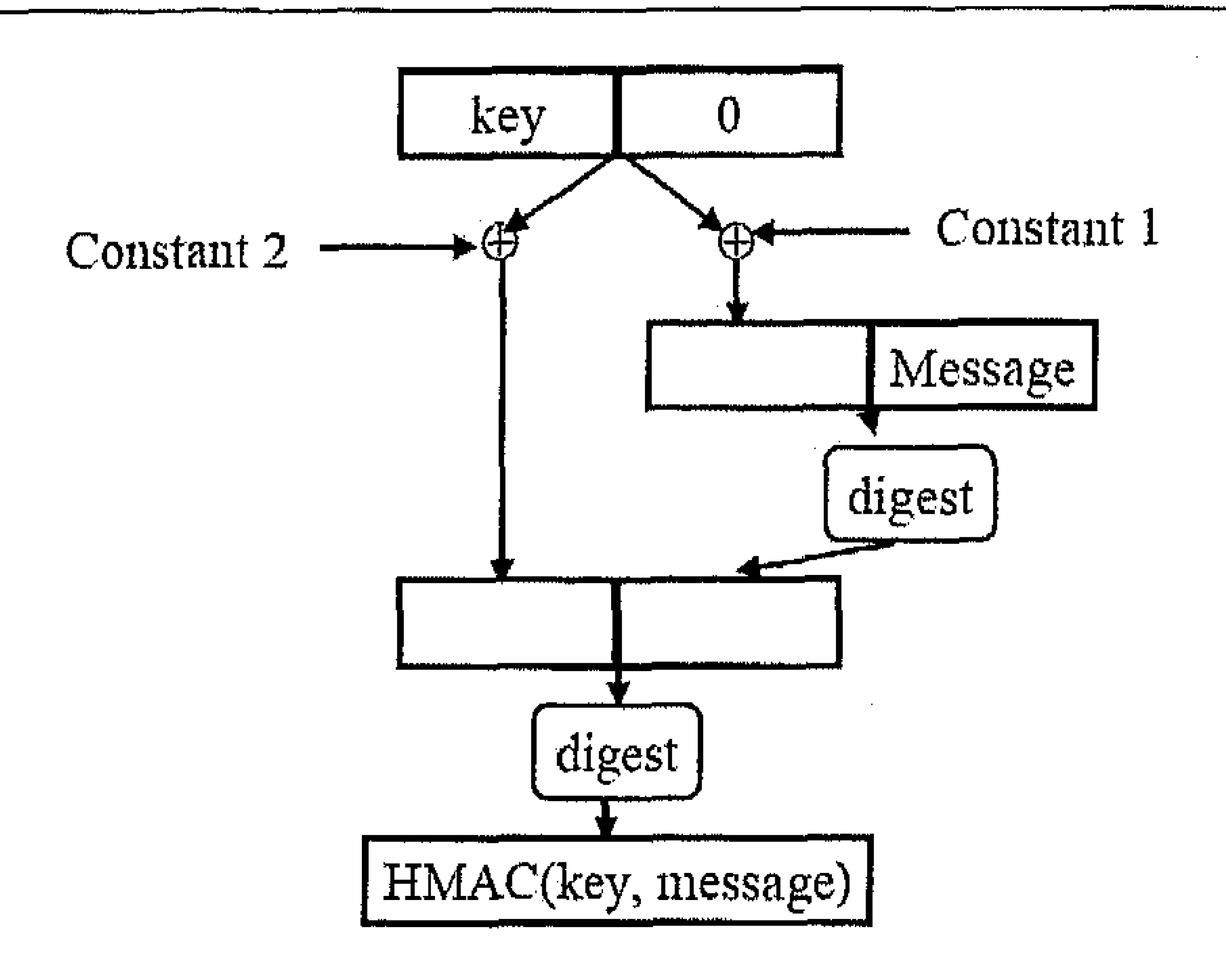
$$\begin{aligned} MD(k|m') = MD(k|m|y) = MD(k|m) + MD(y) \\ = d + MD(y) = d' \end{aligned}$$

فلا يشعر عبيد بتغيير الدخيل للرسالة؛ لأن البصمة التي حسبها متساوية مع البصمة التي وصلته، ويعتبرها أرسلت من زيد، لسد هذه الثغرة نقوم بحساب القيمة التالية:

$$MD(K|MD(K|m))$$

وهكذا لا يمكن أن يضيف الدخيل شيئًا للرسالة، لا عن يمين الرسالة ولا عن يسارها، ولا في الوسط، ولا أن يتقمص شخصية زيد، لأن هذه البصمة محسوبة على بصمة الرسالة لا على الرسالة نفسها. تقوم خوارزمية SHA-1 بحساب هذه البصمة وتسمى HMAC أي (Hash-based MAC) كما هو موضح بالشكل 6.3

شكل 6.3 طريقة توليد بصمة HMAC.

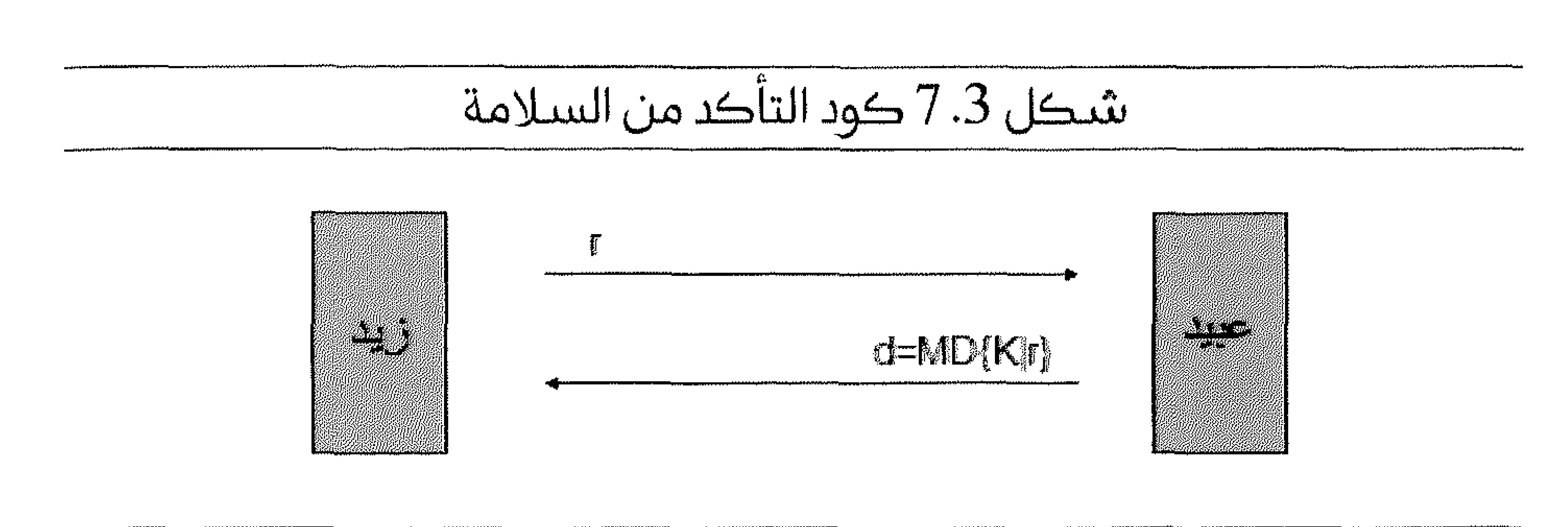


تأخذ الخوارزمية مفتاحًا ذا حجم متغير، ورسالة ذات حجم متغير أيضًا، وتنتج بصمة على 160 بت. ويقع ملء المفتاح ببت صفر حتى يبلغ حجم 512 بت، وكذلك ندمج ثابتين مختلفين لحساب HMAC.

أخيرًا وإن كان MAC أو HMAC تمكنا من التأكد من هوية الرسالة، إلا أنها لا تمكننا من خدمة عدم إنكار الإرسال إذا استعملنا المفتاح السري المشترك، أي إن زيدًا يمكن أن يزعم أنه لم يرسل لعبيد الرسالة، ويدعي أن عبيدًا ولد هو بنفسه لنفسه الرسالة، إذ إنهما يشتركان في المفتاح السري. والهجوم الثاني: أن الدخيل ربما سحب الرسالة من الشبكة، وأعاد إرسالها مرة أخرى في وقت آخر باسم زيد، فيعبر عبيد أن الرسالة جاءته من زيد مرة أخرى، ولا يتفطن إلى أنها من قبل دخيل ما.

3.4 التأكد من هوية المرسل

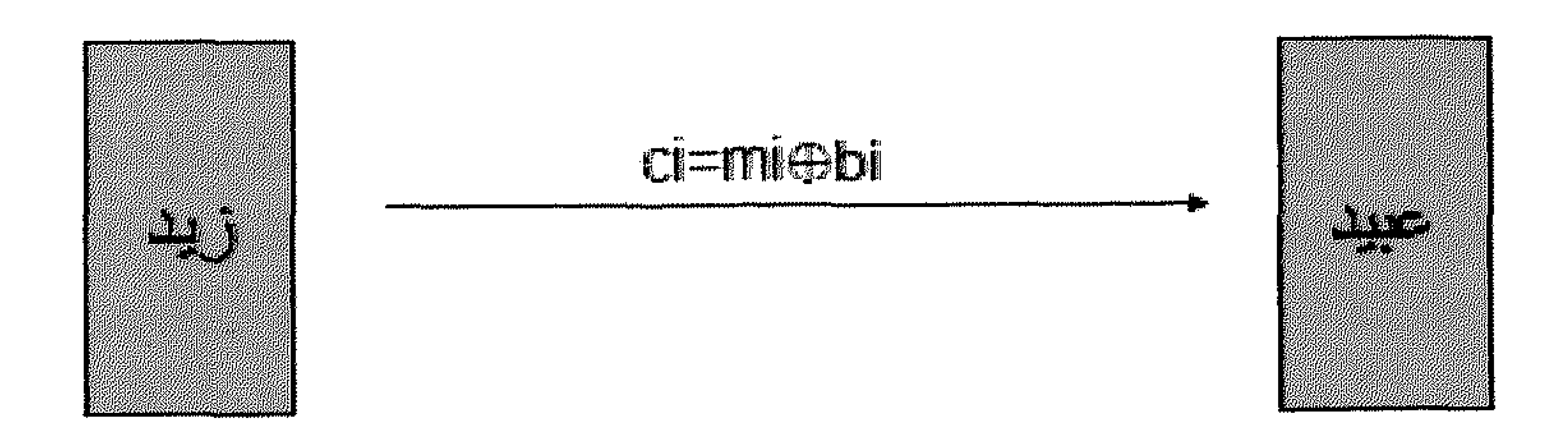
شكل 7.3 كود التأكد من السلامة



يمكن استعمال خوارزميات توليد البصمة من التأكد من هوية متصل ما، وذلك باستعمال بروتوكول التحدي. فإن ادعى شخص لزيد أنه عبيد فإن زيدًا يتحداه بإرسال قيمة عشوائية r يولدها ويحتفظ بها عنده، وينتظر من المدعي إرسال بصمة المفتاح المشترك بينهما مضافًا لقيمة التحدي r . وعندما تصل الرسالة لزيد يقوم بحساب البصمة على المفتاح المشترك بينه وبين عبيد مضافًا للقيمة r التي أرسلها آنفًا، فإن تساوت مع البصمة التي استقبلها فالمدعي فعلاً عبيد، وإلا فلا. وهكذا يكون زيد قد تأكد من هوية عبيد.

3.5 التأكد من السرية

شكل 8.3 كود التأكد من السلامة.



يمكن استعمال خوارزميات توليد البصمة في تشفير البيانات أيضًا، وذلك بصناعة دفق تشفير (one time pad) يعتمد على المفتاح المشترك

بين المتخاطبين على الشبكة. يقوم زيد بحساب بصمة المفتاح المشترك $b_1 = MD(k)$ ثم بحساب بقية البصمات

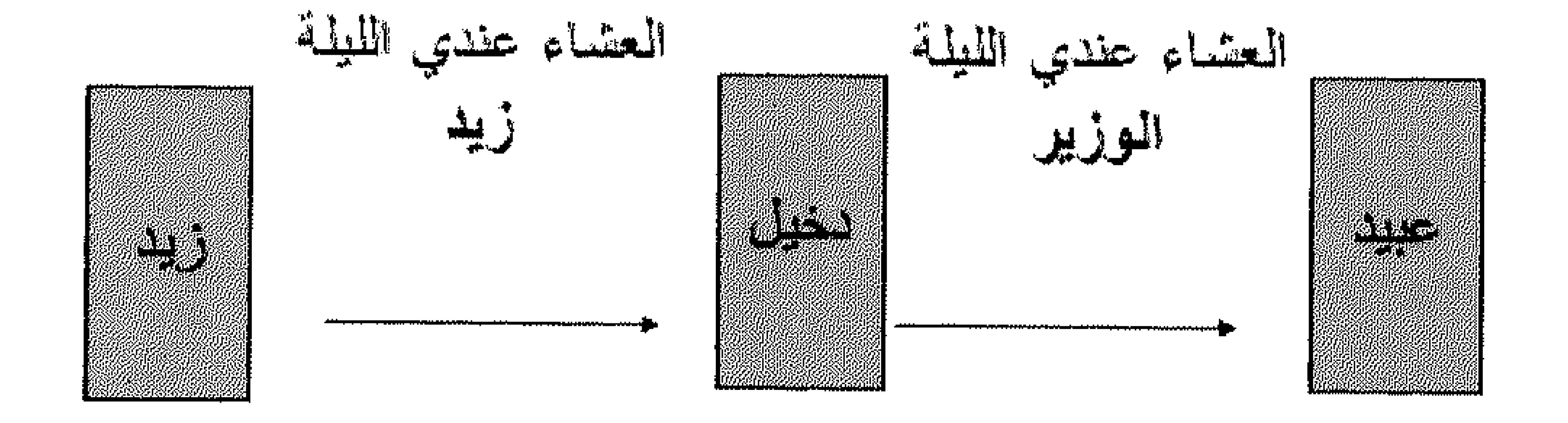
لتشفير البيانات يقوم زيد بعملية xor مع الدفق المولد آنفًا، ويرسل الناتج إلى عبيد يقوم عبيد لفك التشفير بصناعة نفس الدفق؛ لأنه يمتلك المفتاح أيضًا، ويقوم هو بدوره بعملية xor مع النص المشفر ليحصل على النص الواضح للرسالة.

تستعمل خوارزميات توليد البصمة أيضًا في حفظ كلمات السر في نظام يونكس المشهور. إذ يستعمل خوارزمية DES' (نسخة مغايرة قليلاً من خوارزمية DES لئلا يقع كسرها كما كسرتDES) لحساب بصمة كلمة سر ما؛ ليحتفظ بها في بعد في ملف كلمات السر. تولد DES' مفتاحها من كلمة السر فتأخذ 7بت من كل حرف من الأحرف الثمانية الأولى من كلمة السر؛ لتحصل على مفتاح طوله 56 بت. ثم تولد رقمًا عشوائيًا لكل كلمة سر بطول 12 بت يسمى salt، ويحتفظ به مع كلمة السر. يستعمل salt لمنع هجوم على كلمات السر باستعمال القواميس، وأيضًا في تغيير عملية توسيع البيانات في خوارزمية DES. النسخة المبدلة من DES تستعمل في تشفير الثابت 0 باستعمال المفتاح السري، والناتج يحتفظ به مع الsalt ككلمة سر ببصمتها للمستخدم.

4 التوقيع الالكتروني

يسعى التوقيع الالكتروني إلى حل مشكلة إثبات مصدر الرسالة. أي كيف نعرف أو نثبت أن الرسالة أرسلت من قبل شخص ما؟

شكل 9.3 هجوم الإخلال بمصدر الرسالة.



قلنا في الفقرة 2.3 إن المفتاح السري لا يصلح في إثبات مصدر الرسالة، وذلك لأن المفتاح مشترك بين شخصين، فيحتمل أن يكون مرسل

الرسالة واحدًا منهما فلا نستطيع الجزم بمصدر الرسالة. أما في نظام المفتاح العام فإن هذا ممكن، لأن المفتاح الخاص لا يملكه إلا شخص واحد، فلا يمكن لأحد أن ينكر إرسال شيء شفر بمفتاحه الخاص.

4.1.1 متطلبات التوقيع الالكتروني

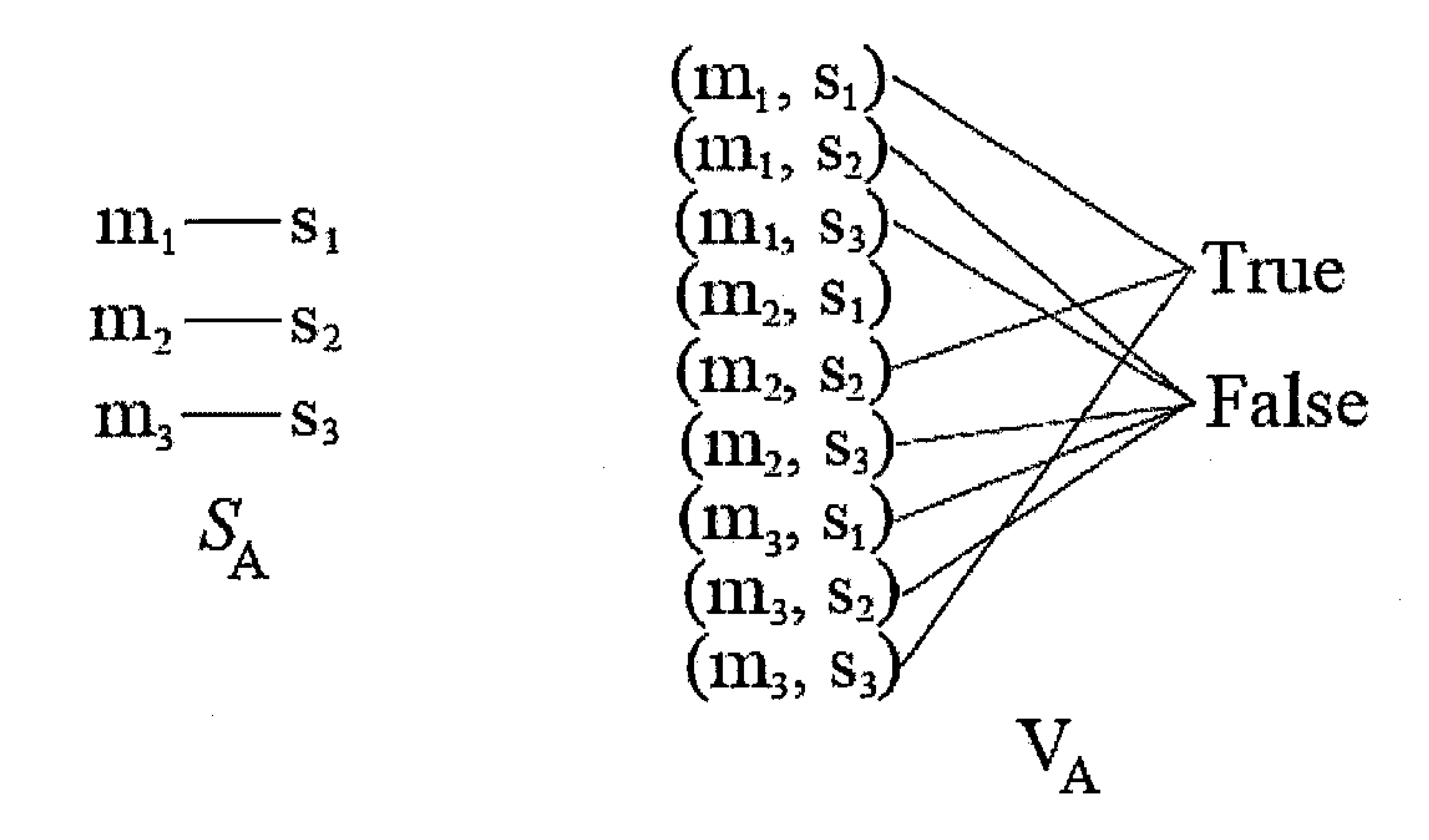
التوقيع الالكتروني يمثل الحجر الأساس في خدمة التأكد من الهوية، وخدمة إثبات مصدر الرسالة.

التمثيل الرياضي:

لتكن

- M مجموعة الرسائل التي يمكن إمضاؤها.
- S مجموعة التوقيعات أو التواقيع (مثال سلسلة من n بت).
- $S_A : M \to S$ دالة التوقيع لمتخاطب ما A الذي يحتفظ بها كسر.
- $V_A : M * S \to \{true, false\}$ دالة التثبت من توقيع A وهي معلومة للعموم.
- $S_A, V_A \Leftarrow$ تمثل مخطط التوقيع الالكتروني لـ A (انظر إلى الشكل 10.3)

شكل 10.3مخطط التوقيع الالكتروني.



إجراءات التوقيع الالكتروني:

- إجراء التوقيع: يقوم زيد ونرمز له بـ A بتوليد توقيع إلكتروني لرسالة $m \in M$ بحساب $s = S_A(m)$ ثم يرسل الزوج (m, s)
- إجراء التثبت: يقوم عبيد بالتثبت من توقيع زيد للزوج (m, s) بحساب $u = V_A(m, s)$ وما يصادق على صحة التوقيع إلا إذا كانت $u = true$.
- متطلب السرية: أن يكون من الصعب لأي شخص غير A أن يجد، لكل $m \in M$ توقيع إلكترونيا $s \in S$ بحيث يكون $V_A(m, s) = true$.

4.1.2 سبل تنفيذ التوقيع الالكتروني

- يمكن أن نعتمد في تنفيذ التوقيع الالكتروني على نظم التشفير بالمفتاح العام القابلة للانعكاس.
- لنفترض أن $E_e = M \to C$ دالة التشفير بالمفتاح العام. لنفترض أيضًا أن $M = C$. فلو اعتبرنا D_d دالة فك التشفير بالمفتاح الخاص لـ E_e،ونظرا أن كلتا الدالتين هي عمليات إبدال فإنه $D_d(E_e(m)) = E_e(D_d(m))$ لكل m $m \in M$. كل طريقة تشفير بالمفتاح العام من هذا النوع، فإنها تسمى قابلة للانعكاس.
- والآن لصناعة مخطط للتوقيع الإلكتروني نفترض $M = C$ وزوج مفتاح تشفير (e, d) فنعرف التوقيع S_A ب D_d أي $s = D_d(m)$ ونعرف التثبت من التوقيع V_A ب

$$V_A(m, s) = \begin{cases} true, & if E_e(s) = m, \\ false, & otherwise \end{cases}$$

- لكن هذا المخطط قابل للتزوير وذلك بأن:
- يختار الدخيل توقيعا عشوائيًا $s \in S$ ويحسب $m = E_e(s)$
- بما أن $S = M$ فيمكنه أن يرسل (m, s) كرسالة مع إمضائها الإلكتروني.
- وفي إجراء التثبت سنحصل على $true$ مع أن A لم يوقع الرسالة m

يكمن الحل في جعل جزء من مجموعة M في محتواة M' تمثل مجموعة الرسائل القابلة للتوقيع فقط ونعيد تعريف دالة التثبت V_A $S \to \{true, false\}$ بما يلي:

$$V_A(m,s)=\begin{cases}true, & if E_e(s)\in M',\\ false, & otherwise\end{cases}$$

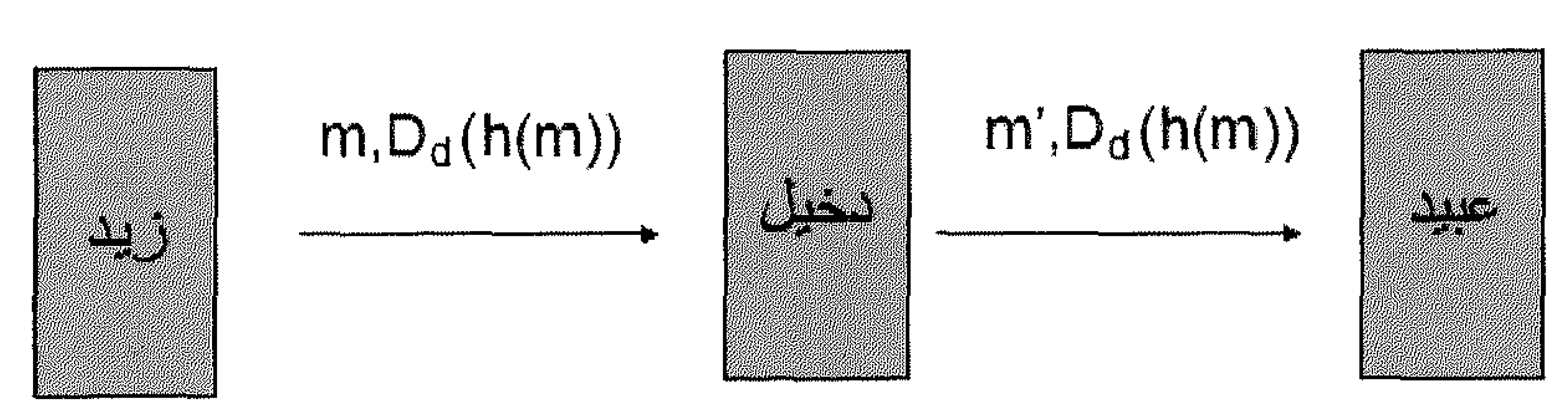
ويمكن استرداد الرسائل إذ أن $m = E_e(s)$.

هذا المخطط المعدل يكون مخططًا آمنًا لو كانت المجموعة M' مجموعة صغيرة بالشكل الكافي من المجموعة M .

4.1.3 التوقيع الالكتروني باستعمال RSA وSHA-1

يمكن استعمال نظام RSA لتنفيذ التوقيع الالكتروني؛ لأنه يلبي خاصية الانعكاس الآنفة الذكر $D_d(E_e(m)) = E_e(D_d(m)) = m$. ونمنع التزوير بتوقيع رسائل ذات هيكلة محددة مثل أن يسمي زيد نفسه (المرسل) في الرسالة، أو أن يرسل بصمة مولدة من قبل خوارزمية توليد بصمات معينة (نستعمل SHA-1 مع RSA). تكون هذه البصمة موقعة إلكترونيًا مع الرسالة، ويتثبت عبيد من البصمة عندما تصل إليه (انظر إلى الشكل 3. 11). كما يمكن تشفير البصمة والرسالة للحفاظ على السرية أيضًا.

شكل 3. 11 إستعمال RSA والبصمة



4.1.4 التوقيع الالكتروني باستعمال مخطط طاهر الجمل

مخطط توقيع الجمل يعتمد على صعوبة حساب اللوغاريثمات المحدودة، وصفها طاهر الجمل في (1984) وهذه الطريقة التي وصفها قليلة الاستعمال في التطبيقات العملية، ولكن جملة من البدائل المعتمدة على فكرة الجمل هي الآن المستعملة في التطبيق.

1 عوامل النظام:

- دالة توليد بصمة صعبة الكسر.
- pعدد أولي كبير حيث يكون حساب اللغاريثمات المحددة $mod\ p$ صعبة.
- g مولد عشوائي مختار لتوليد الأعداد على زمرة Z_p^*
- هذه العوامل معلنة للجميع.

2 توليد المفتاح:

- نختار مفتاح سري xبحيث يكون $1 < x < p - 1$
- نحسب $y = g^x mod\ p$
- المفتاح العام هو (p, g, y)
- المفتاح السري هو x
- هذه الخطوات يقوم بها الموقع مرة واحدة.

3 توليد التوقيع:

لتوليد التوقيع الالكتروني لرسالة m يقوم الموقع بالخطوات التالية:

يختار رقما عشوائيا k بحيث يكون:

- $1 < k < p - 1$ و $\gcd(k, p - 1) = 1$
- يحسب القيمة $r \equiv g^k \ (mod\ p)$
- يحسب القيمة$s \equiv (H(m) - xr)k^{-1} \ (mod\ p)$
- لو كان $s = 0$ يكرر الخطوة السابقة.
- الزوج (s, r) هو التوقيع الالكتروني للرسالة m والموقع أو الموقع يقوم بهذه الخطوات في كل مرة يريد توقيع رسالة ما.

4 التثبت من التوقيع:

نقوم بالتثبت من صحة التوقيع الالكتروني للزوج (s, r) للرسالة m بما يلي:

- $0 < r < p \ \ and \ \ 0 < s < p - 1$
- $g^{\mathrm{H(m)}} \equiv \mathrm{y^r r^s} \ (mod\ p)$

المتثبت يقبل كل توقيع توافقت فيه كل الشروط ويرفض إذا اختل شرط واحد.

إثبات صحة الخوارزمية:

تكون خوارزمية التوقيع صحيحة بمعنى أن كل توقيع مولد بهذه الخوارزمية يقع قبوله دائمًا من المتثبت.

عندنا $H(m) \equiv xr + sk \pmod{p-1}$ وبالتالي فإن:

$$g^{H(m)} \equiv g^{xr} g^{ks} \equiv (g^x)^r (g \quad) k^s \equiv (y)^r (r)^s \pmod{p}$$

لكي يستطيع الدخيل أن يزور توقيعا مولدا بمخطط الجمل فليس له إلا أن يكتشف المفتاح الخاص أو يكسر خوارزمية توليد البصمة. وهاتان المسألتان يعتقد أنهما صعبتان جدًا.

4.1.5 التوقيع الالكتروني المعياري DSS

طورت NIST خوارزمية التوقيع الالكتروني DSA معتمدة على مخطط الجمل لتصبح بعد ذلك الخوارزمية المعيارية DSS سنة 1991 ثم عدلت قليلا عام 1996 ثم عام 2000 ثم عام 2009.

1 عوامل النظام:

- H دالة توليد بصمة تشفيرية وعادة ما نستعمل SHA-1 ولكن SHA-2 أقوى ويمكن استعمالها الآن مع DSS. ويمكن أن تقسم البصمة المتولدة لتوافق حجم مفتاح التشفير.
- الاتفاق على طول العددين L و N ففي النسخة الأولى لـDSS كانت تفرض أن يكون H من مكررات 64 فيما بين 512 و1024. وتنصح NIST بطولي 2048 أو 3072 للسنوات فيما بعد 2010 و2030 حيث تستعمل حجمًا أطول ل N فيكون زوج (L,N) مساوية إما ل(1024,160)،(2048,256)(2048,224)،و(3072,256).
- اختيار رقم أولي q بحجم N بت حتمًا لا بد أن يكون أقل أو مساويًا لطول البصمة المولدة.
- اختيار رقم أولي بحجم L بت $mod\ p$ بحيث تكون $p-1$ من مكررات q .
- اختيار g بترتيب $mod\ p$ مساويا ل q . يمكن أن نحصل على ذلك بوضع $g = h^{(p-1)/q} mod\ p$ لرقم ما $h, (1 < h < p-1)$ ونعيد

الكرة لو تحصلنا على h اختيارات أكثر.واحدًا النتيجة كانت لو h تعطينا قيمة g قابلة للاستعمال وعادة ما نختار $h = 2$.

- القيم (p, q, g) يمكن أن تكون مشتركة بين مجموعة مختلفة من المستخدمين للنظام.

2 توليد المفتاح:

1. نختار مفتاحًا سريًا x بحيث يكون $0 < x < q$
2. نحسب $y = g^x mod\ p$
3. المفتاح العام هو (p, q, g, y)
4. المفتاح السري هو x
5. هذه الخطوات يقوم بها الموقع مرة واحدة وتوجد خوارزمية فعالة في حساب الأس سبق التعرض إليها.

3 توليد التوقيع:

لتوليد التوقيع الالكتروني لرسالة m يقوم الموقع بالخطوات التالية:

1. يختار رقمًا عشوائيًا k بحيث يكون $\boldsymbol{1 < k < q}$
2. يحسب القيمة $\boldsymbol{r = (g^k\ mod\ p)\ mod\ q}$
3. يحسب القيمة $\boldsymbol{s = \left((H(m) + xr)k^{-1}\right) mod\ q}$
4. لو كان $\boldsymbol{r = 0}$ أو $\boldsymbol{s = 0}$ يكرر الخطوة السابقة.
5. الزوج (s, r) هو التوقيع الالكتروني للرسالة m والموقع أو الموقع يقوم بهذه الخطوات في كل مرة يريد توقيع رسالة ما. كما يمكن استعمال خوارزمية اقليدس المعممة، التي سبقت في فصل مقدمة إلى علم التشفير لحساب $k^{-1}mod\ q$

4 التثبت من التوقيع:

- يقع رفض التوقيع لو أخل بأحد هذين الشرطين:

$$\boldsymbol{0} < r < q \quad or \quad 0 < s < q$$

- نحسب القيمة التالية: $\boldsymbol{w = (s)^{-1}mod\ q}$
- ثم القيمة التالية: $\boldsymbol{u1 = (H(m) * w)\ mod\ q}$
- ثم القيمة التالية: $\boldsymbol{u2 = (r * w)\ mod\ q}$
- ثم القيمة التالية: $\boldsymbol{v = \left((g^{u1} * y^{u2}) modp\right)\ mod\ q}$

- ويكون التوقيع صحيحًا إذا كان $v = r$

كما هو ملاحظ فإن DSA مشابه لخوارزمية طاهر الجمل.

5 إثبات صحة الخوارزمية :

تكون خوارزمية التوقيع صحيحة، بمعنى أن كل توقيع مولد بهذه الخوارزمية يقع قبوله دائمًا من المتثبت.

أولاً: إذا كان عندنا $g = h^{(p-1)/q} mod\, p$ ، فإنه وباستعمال قاعدة Fermat الصغيرة سيكون عندنا $g^q \equiv h^{p-1} \equiv 1\ (mod\, p)$.

وبما أن $g > 1$ و q عدد أولي فإن g ذات ترتيب q .

الموقع يحسب $s = (H(m) + xr)k^{-1})\ mod\ q$ وإذن

$$k \equiv H(m)s^{-1} + xrs^{-1} \equiv H(m)w + xrw\ (mod\ q)$$

وبما أن g ذات ترتيب q فيكون عندنا

$$r = (g^k mod\ p) mod\ q = (g^{u1} y^{u2}\ mod\ p)\ mod\ q = v$$

ونثبت صحة الخوارزمية بالآتي:

$$r = (g^k mod\ p) mod\ q = (g^{u1} y^{u2} mod\ p) mod\ q = v$$

5 إدارة المفاتيح و البنية التحتية للمفتاح العامPKI

إن التشفير التماثلي والتشفير بالمفتاح العام يحققان سرية البيانات، شرط أن يكون هناك بنية تحتية آمنة لتوزيع المفاتيح. ومع أن التشفير بالمفتاح العام خفض كثيرا عدد المفاتيح التي يجب إدارتها، إلا أنه ما زالت هناك حاجة لإيجاد قناة ذات هوية موثوقة لتوزيع المفاتيح العامة. هذه القناة ستجعل من الخدمات الأمنية – آنفة الذكر – كالتأكد من هوية الرسالة، والمرسل، والتوقيع الالكتروني خدماتٍ موثوقةً وآمنةً. ومن هنا تنبع أهمية تطوير وتصميم الآليات والبنية التحتية التي تدعم إيجاد مثل هذه القنوات للتطبيقات والبيئات الرقمية.

إدارة المفاتيح تهتم بأمور ثلاثة:

1. توزيع مفاتيح التشفير.
2. آليات الربط بين هوية كيان ما مع مفتاح التشفير.
3. توليد ومتابعة وإلغاء مفاتيح التشفير هذه.

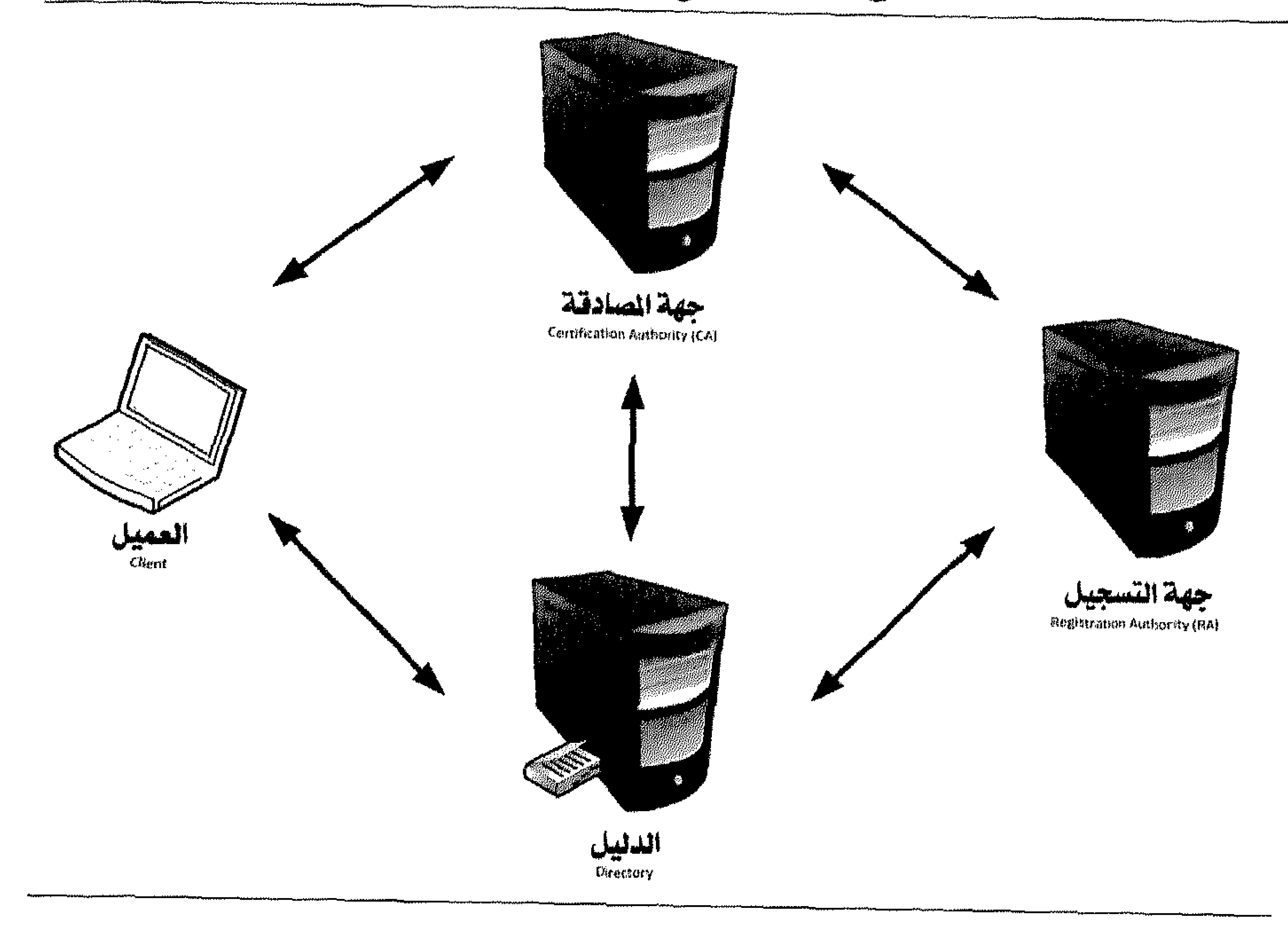
نصب اهتمامنا في الفقرات اللاحقة على البنية التحتية للمفتاح العام PKI والإجابة عن سؤالين أساسيين وهما: كيفية الربط بين المفتاح مع هوية كيان ما؟ وكيف يمكن أن نمثل هوية هذا الكيان (naming)؟

قلنا: إن PKI يمكن المستخدمين من الربط بين مفتاح ما والكيان الذي يملكه. للانضمام لـPKI يولد زيد مفتاحه الخاص والعام، ثم يتصل بجهة تصديق الشهادات (certificate Authority (CA وهي جهة موثوقة تقوم بإصدار والمصادقة على الشهادات، معلمًا لها أنه زيد، وأن المفتاح العام هو مفتاحه، تقوم CA بالتثبت من صحة ادعاء زيد امتلاكه المفتاح، ثم إن ثبت لديها ذلك تقوم بتوقيع شهادة رقمية تقرر "أن المفتاح k_A يملكه زيد". وهكذا نُكَوِّنُ ثقة متبادلة بين زيد وعبيد عن طريق جهة المصادقة الموثوقة، فعبيد يمكنه الآن التأكد من خلال الشهادة للحصول على مفتاح زيد واعتباره صحيحًا، وكذلك زيد يقوم بنفس الشيء للحصول على مفتاح عبيد من خلال شهادته المصدقة من طرف جهة المصادقة CA.

5.1 عمارة البنية التحتية PKI

تتمثل خدمات البنية التحتية في إصدار الشهادات، أي الربط بين الكيان ومفتاحه، وأيضًا إدارة المفتاح من إلغاء واسترداد وتحديث. تتكون عمارة البنية التحتية من أربع عناصر أساسية، وهي: جهة المصادقة Certification Authority (CA) وعنصر الدليلDirectory، وجهة التسجيل Registration Authority (RA) وأخيرًا العميل Client. كل هذه العناصر ترتبط وتتصل ببعض، ولكن كل عنصر منها له مهام محددة (انظر إلى الشكل 12.3).

شكل 12.3 عمارة البنية التحتية PKI



1. مهام جهة المصادقة:

- توليد الشهادات ونشرها في الدليل.
- صيانة قائمة الشهادات الملغاة CRL (Certification Revocation List) في الدليل. يقع التحقق من القائمة من طرف العملاء، أو من خدمة التحقق من صحة الشهادة.
- نسخ احتياطي لبعض المفاتيح لأغراض الاسترداد أو الضمان لاحقًا.

2. مهام الدليل:

- يجعل الشهادات وقائمة الشهادات الملغاة متاحة وسهلة التناول للجميع.
- تعريف المستخدمين وتحديدهم بشكل فريد (أوحد)، ولهذا يحتاج إلى بيانات المستخدمين الدقيقة والمحدثة
- يجب أن تكون خاصية التوفرية فيه عالية (متاح عندما يحتاج إليه)

3. مهام جهة التسجيل:

- إدارة عملية تسجيل المستخدمين وإصدار الشهادات.
- ضمان صحة هوية المستخدم.

4. استعمالات العميل:

- يستخدم العميل البنية التحتية في عدة خدمات منها: (1) التأكد من الهوية (في اتجاه واحد أو اتجاهين أو ثلاثة اتجاهات) (2) في تشفير البيانات (3) في توقيع الوثائق، والصفقات الإلكترونية).
- التطبيقات التي تدعم البنية التحتية، وكذلك النظم مثل **CAPI** وهي جملة من الدوال التي تدعم التشفير في نظام ويندوز، وأيضًا في التطبيقات الخاصة.

وبصفة عامة هناك نوعان من البنى التحتية للمفتاح العام: مفتوحة ومغلقة. فالبنية المفتوحة تستخدم بين الشركات والمنشآت على نطاق واسع. والبنية المغلقة تكون مقتصرة عل مجموعة معينة من المستخدمين، كمستخدمي الشبكات الافتراضية مثلاً.

5.2 شهادات المصادقة

تعتبر الشهادة علامة ودليلاً تربط بين كيان ما ومفتاحه. لنفترض أن جهة المصادقة C وقعت شهادة مصادقة لزيد ونرمز له ب A تربطه بمفتاحه k_A مع طابع أو ختم زمني $timestamp(T)$

يتحقق عبيد من شهادة زيد ليحصل على مفتاح زيد ويعتبره صحيحًا، ولكن عبيد يجب عليه التحقق من صحة مفتاح الجهة المصادقة C للتحقق من صحة الشهادة. هناك طريقتان في التعامل مع هذه المسألة:

1. نزع توقيع الجهة المصادقة C تمامًا.
2. هيكلة الشهادة ضمن سلسلة تواقيع.

5.2.1 مخطط شجرة ماركلي للتأكد من الهوية Merkles's Tree

المخطط يندرج ضمن الحل الأول وهو نزع توقيع الجهة المصادقة C تمامًا. تكمن الفكرة الأساسية في أن الشهادات يمكن أن يحتفظ بها كبيانات في ملف فأي تغيير يحدث على الشهادات سيغير الملف تلقائيًا. وهذا يرجع معالجة الشهادات المزورة إلى مسألة التأكد من سلامة البيانات، ولهذا نستعمل دوال توليد البصمة التي تمكننا من اكتشاف التغيير الحاصل على ملف الشهادات.

التمثيل الرياضي:

- لتكن $Y_1, \dots, Y_2$ حيث كل Y_i زوج مكون من هوية كيان ومفتاحه العام. يحتفظ بهذه الأزواج في ملف ما.
- لتكن $f: D * D \to D$ حيث تمثل D مجموعة من سلاسل مكونة من بتات ولتكن $h: N * N \to D$ دالة توليد بصمة تشفيرية.

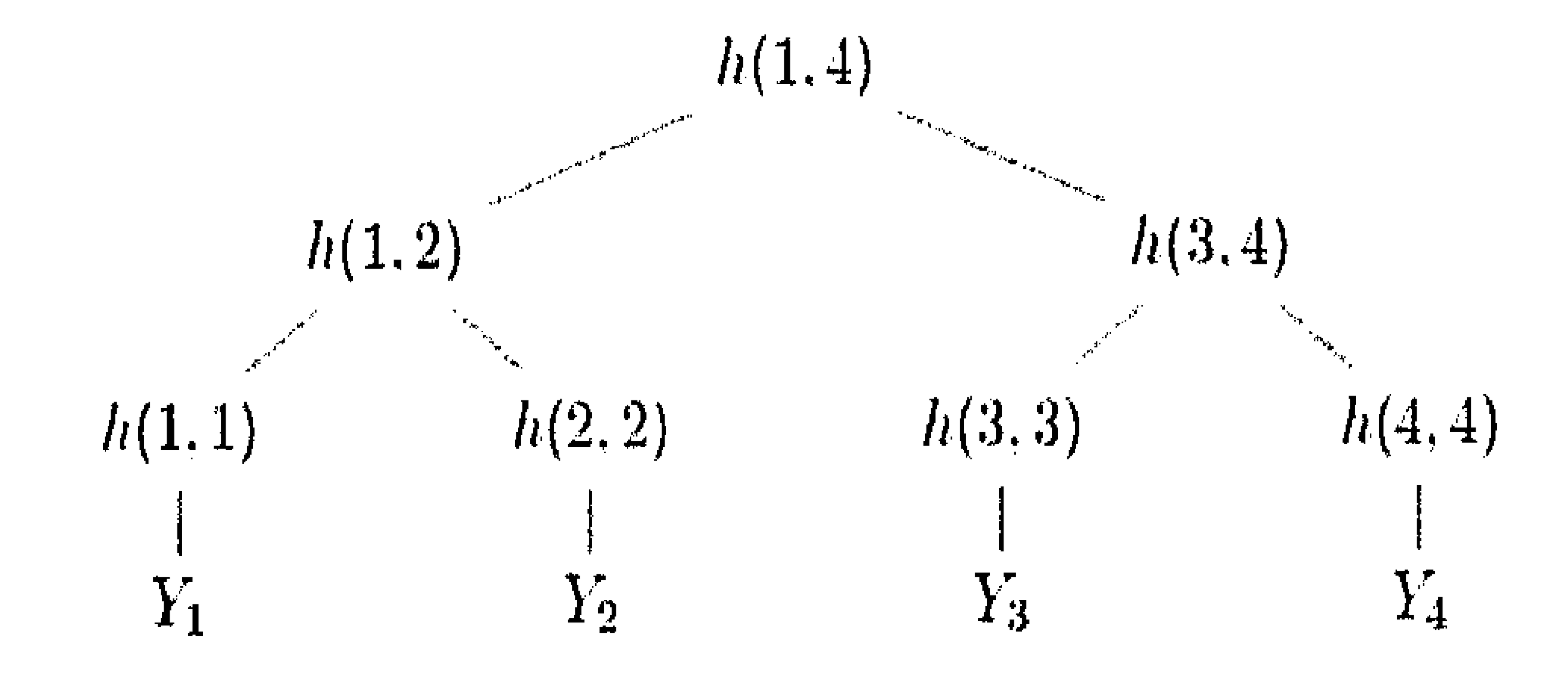
$$h(i,j) = \begin{cases} f\left(h\left(i, ceiling\left(\frac{i+j}{2}\right)\right), h\left(ceiling\left(\frac{i+j}{2}\right) + 1, j\right)\right) if\ i < j \\ f(C_i, C_j)\ otherwise \end{cases}$$

- الدالة **ceiling** تعطينا أقرب عدد طبيعي أقل من العدد العشري الذي تأخذه.

تسمى بصمة الملف ككل الجذر وهي $h(1, n)$.

المناداة المتكررة للدالة (recursion) تولد هيكل شجرة البصمات (انظر إلى الشكل 13.3 لملف يحوي أربع أزواج (هوية كيان،مفتاح عام))

شكل 3. 13مثال لشجرة ماركلي



بصمة الملف ككل هي $h(1,4)$ وهي الجذر، وهي معلومة من طرف جميع المستخدمين للنظام.

مثال: للتحقق منY_3 ، يعيد زيد حساب الجذر $h(1,4)$ أي بصمة الملف ككل. وهذا يتطلب حساب كل عقد الشجرة التي هي في الطريق من Y_3 إلى $h(1,4)$ أي:

$$\boldsymbol{h(3,3) = f(Y_3, Y_3)}$$
$$\boldsymbol{h(3,4) = f\big(h(3,3), h(4,4)\big)}$$
$$\boldsymbol{h(1,4) = f\big(h(1,2), h(3,4)\big)}$$

وهذا يعني أنه إما البصمات أو الشهادات نفسها متاحة. وللفاعلية تحسب البصمات مسبقًا. هذه القيم الوسيطة والمحسوبة آنفًا للتحقق من الشهادة تسمى مسار المصادقة. ففي المثال يكون مسار المصادقة لY_3 القيم$h(4,4)$ و$h(1,2)$. مسار المصادقة هذا هو الذي يمثل الشهادة C_3.

مخطط ماركلي لا يتطلب إلا أن تكون قيمة الجذر معلومة من الجميع، والملف متاح للعموم. إذا تم المساس بأي زوج (هوية كيان،مفتاح عام) فقيمة الجذر ستتغير وسيكتشف هذا خلال عملية التحقق من صحة الشهادة، لكن بالمقابل لو غير كيان ما مفتاحه العام فلا بد من إعادة حساب الجذر وتوزيعه من جديد على مستخدمي النظام.

إن مخطط ماركلي يعالج مسألة الشهادات المنظمة بشكل هرمي ويقترح آلية لا تستعمل التوقيع بالمفتاح العام لتوليد الشهادات ولكن الحاجة إلى أن يكون ملف الشهادات عاما لتمكن من إعادة حساب الجذر

تجعل هذا المخطط غير عمليا للشبكات التي تنطوي على مجموعة كبيرة جدًا من الشهادات في الأنظمة واسعة النطاق.

5.2.2 سلسلة تواقيع الشهادة

الشكل المعتاد للشهادة هو $C = \{K_A, A, T\}_{k_C^{-1}}$ كما سبق الإشارة إلى ذلك. فالمصدر للشهادة يستعمل مفتاحه الخاص لتشفير بصمة تأكيد هوية حامل الشهادة مع معلومات حول تاريخ إصدار الشهادة، وانقضاء صلاحيتها. للتحقق من الشهادة يحصل المتحقق على المفتاح العام للجهة المصدرة للشهادة ويفك تشفير الشهادة ليحصل على البصمة ويتحقق من بيانات الشهادة. إن كان المصدر للشهادة يملك هو بدوره شهادة فحينئذ يمكن للمتحقق من الحصول على مفتاح الجهة المصدرة، وهذا ينقل المسألة إلى مسألة في مستوى آخر: كيف يمكن أن نتحقق من شهادة جهة مصدرة للشهادات؟

هناك طريقتان للتعامل مع هذه المسألة:

1. صناعة شجرة هرمية حيث يكون المفتاح العام لجذر الشجرة معلوم للجميع بطريقة خارجية مستقلة (تبادل مباشر يدًا بيد مثلا).
2. أو السماح بترتيب أو اتفاق اعتباطي بين المصدرين للشهادات، بالاعتماد على المعرفة الفردية لكل جهة إصدار بالجهات الأخرى.

لتوضيح هذه الطرق والشهادات وعملية إصدارها نعرض في الفقرات الموالية لمعيار X.509.

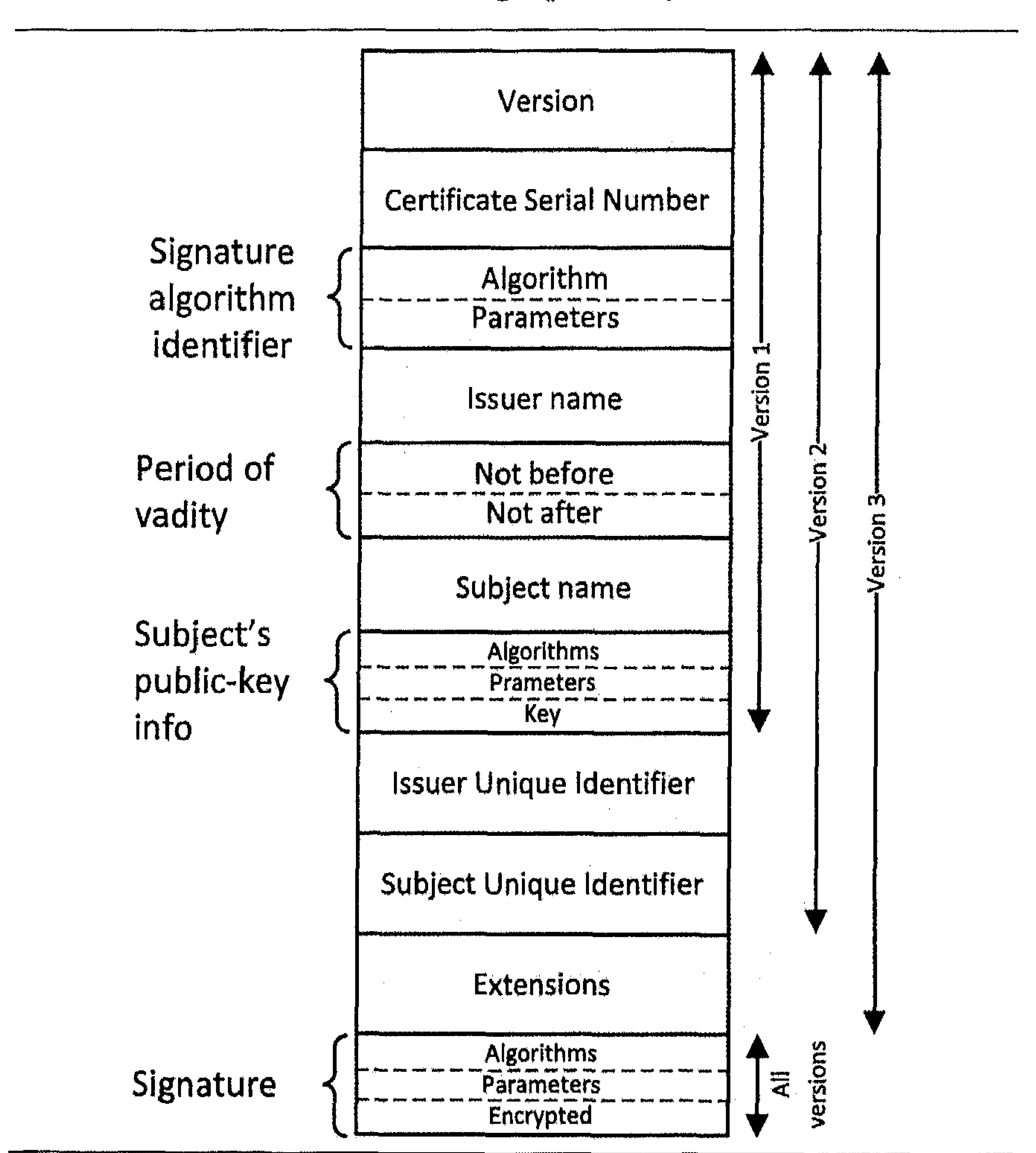
5.2.3 معيار X.509

هذا المعيار هو جزء من سلسلة معايير التوصيات X.500 التي أخرجها الإتحاد العالمي للاتصالات ITU-T حيث تحدد الإطار العام لتوفير خدمة التحقق من الهوية وغيرها. وتستعمل توصيفات الشهادات، وطرق التحقق من صحتها، وبروتوكلات التحقق من الهوية المضمنة في X.509 في مختلف المجالات والتقنيات مثل IPSec,SSL/TLS,SET وS/MIME. يعتمد X.509 على التشفير بالمفتاح العام، ويوصي باستعمال خوارزمية RSA، دوال توليد البصمة والتوقيع الالكتروني. تم إصدار النسخة الأولى في يوليو (1988) لتصدر النسخة الثالثة من X.509v3 عام (1995) ومراجعتها في عام (2000)، وكثيرًا ما يطلق على X.509v3 اسم PKIX أي بنية المفتاح العام PKI بتوصيف X.509.

هيكل شهادة:X.509

تعتبر شهادة X.509 لب هذا المعيار، فكل مستخدم يحصل على شهادة خاصة به. هذه الشهادات تصدرها جهة موثوقة CA وتضعها هي أو المستخدم في الدليل. ومهمة هذا الأخير تمكين المستخدمين من الوصول إلى هذه الشهادات بشكل سهل وسريع. تتكون الشهادة من عدة عناصر تم إضافتها في النسخة الأولى والثانية والثالثة (انظر إلى الشكل 3. 14) وهي بالتفصيل الآتي.

شكل 3. 14هيكل شهادات X.509



- رقم النسخة **V**:
- رقم تسلسلي **SN**: لا بد أن يكون فريدًا من جملة الشهادات التي تصدرها هذه الجهة، أي لا بد أن يكون الزوج (الجهة المصدرة، الرقم التسلسلي) فريدًا أو أوحدًا.
- معرف بخوارزمية التوقيع **AI**: وكل العوامل المستعملة في التوقيع على الشهادة.
- اسم الجهة المصدرة **CA** المعطى من **X.509** والتي وقعت على الشهادة، ولو تم استعمال هذا الاسم من أكثر من كيان فإن هناك اسمًا فريدًا اختياريًا آخر.
- مدة الصلاحية **T** وهي مدة صلاحية استعمال الشهادة.
- اسم المستخدم **A** الذي سيتم ربطه بمفتاحه العام في الشهادة ولو تم استعمال هذا الاسم من أكثر من كيان فإن هناك اسما فريدًا اختياريًا آخر.
- معلومات مفتاح عام المستخدم **Ap** حيث تحدد الخوارزمية، عواملها والمفتاح العام للمستخدم.
- التوقيع الذي يحتوي على بصمة بقية الحقول مشفرة بالمفتاح الخاص للجهة المصدرة.

يحصل المستخدم A على شهادة من CA هي التالية: (انظر إلى الشكل 15.3)

$$CA \ll A \gg = \{V, SN, AI, CA, TA, A, Ap\}_{K_{CA}^{-1}}$$

شكل 3. 15مثال لشهادة X.509 v3

```
Certificate:
   Data:
       Version: 3 (0x2)
       Serial Number: 1 (0x1)
       Signature Algorithm: md5WithRSAEncryption
       Issuer: C=ZA, ST=Western Cape, L=Cape Town, O=Thawte Consulting cc,
               OU=Certification Services Division,
               CN=Thawte Server CA/emailAddress=server-certs@thawte.com
       Validity
           Not Before: Aug  1 00:00:00 1996 GMT
           Not After : Dec 31 23:59:59 2020 GMT
       Subject: C=ZA, ST=Western Cape, L=Cape Town, O=Thawte Consulting cc,
                OU=Certification Services Division,
                CN=Thawte Server CA/emailAddress=server-certs@thawte.com
       Subject Public Key Info:
           Public Key Algorithm: rsaEncryption
           RSA Public Key: (1024 bit)
               Modulus (1024 bit):
                   00:d3:a4:50:6e:c8:ff:56:6b:e6:cf:5d:b6:ea:0c:
                   68:75:47:a2:aa:c2:da:84:25:fc:a8:f4:47:51:da:
                   85:b5:20:74:94:86:1e:0f:75:c9:e9:08:61:f5:06:
                   6d:30:6e:15:19:02:e9:52:c0:62:db:4d:99:9e:e2:
                   6a:0c:44:38:cd:fe:be:e3:64:09:70:c5:fe:b1:6b:
                   29:b6:2f:49:c8:3b:d4:27:04:25:10:97:2f:e7:90:
                   6d:c0:28:42:99:d7:4c:43:de:c3:f5:21:6d:54:9f:
                   5d:c3:58:e1:c0:e4:d9:5b:b0:b8:dc:b4:7b:df:36:
                   3a:c2:b5:66:22:12:d6:87:0d
               Exponent: 65537 (0x10001)
       X509v3 extensions:
           X509v3 Basic Constraints: critical
               CA:TRUE
   Signature Algorithm: md5WithRSAEncryption
       07:fa:4c:69:5c:fb:95:cc:46:ee:85:83:4d:21:30:8e:ca:d9:
       a8:6f:49:1a:e6:da:51:e3:60:70:6c:84:61:11:a1:1a:c8:48:
       3e:59:43:7d:4f:95:3d:a1:8b:b7:0b:62:98:7a:75:8a:dd:88:
       4e:4e:9e:40:db:a8:cc:32:74:b9:6f:0d:c6:e3:b3:44:0b:d9:
       8a:6f:9a:29:9b:99:18:28:3b:d1:e3:40:28:9a:5a:3c:d5:b5:
       e7:20:1b:8b:ca:a4:ab:8d:e9:51:d9:e2:4c:2c:59:a9:da:b9:
       b2:75:1b:f6:42:f2:ef:c7:f2:18:f9:89:bc:a3:ff:8a:23:2e:
       70:47
```

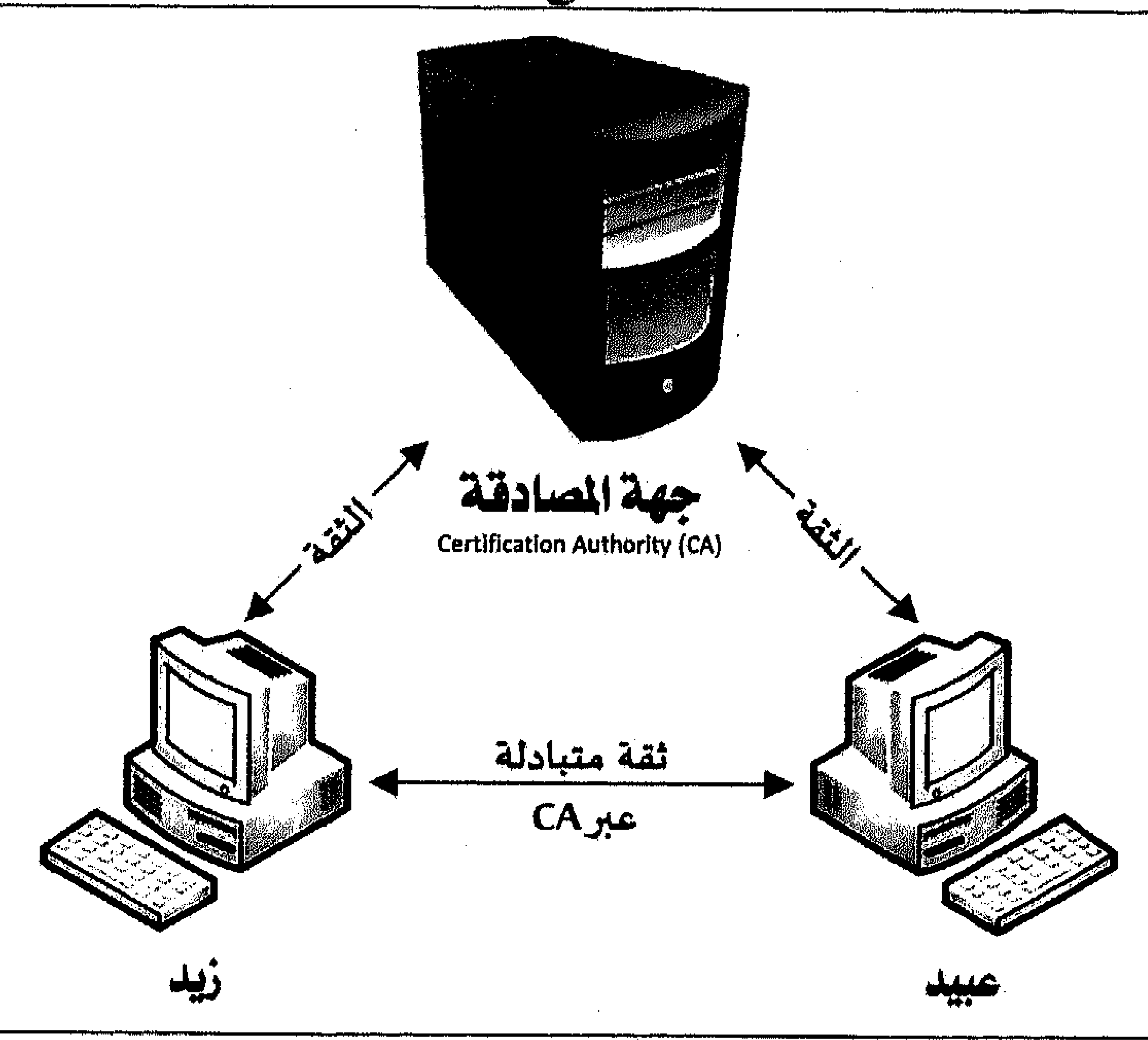
للتحقق من صحة الشهادة $<< A >>$ CA وصحة المفتاح العام المولد، يستخرج زيد المفتاح العام للجهة المصدرة من الشهادة، ويفك تشفير التوقيع الالكتروني. ثم يستعمل المعلومات التي تحصل عليها لإعادة حساب البصمة من خلال الحقول الأخرى. لو تطابقت البصمة مع التوقيع الالكتروني فيعتبر التوقيع صحيحًا، إن كان المفتاح العام للمصدر صحيحًا. ثم ينظر زيد لمدة الصلاحية ليضمن أن الشهادة حديثة.

5.3 نماذج الثقة Trust Models

هناك جملة من نماذج الثقة التي تملي على المستخدم كيفية التحقق من صحة الشهادة أهمها ثلاث وهي:

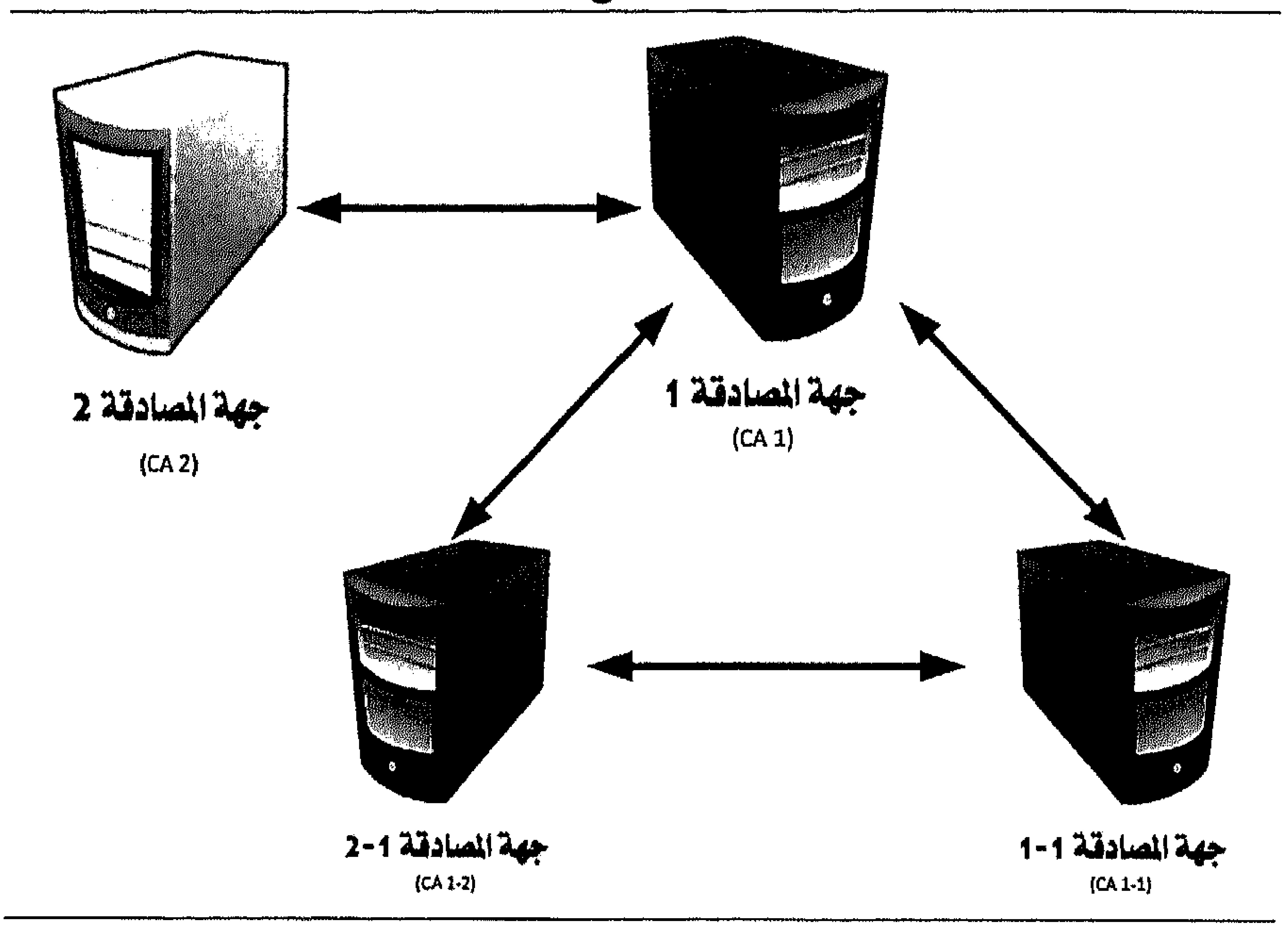
نموذج الثقة المباشرة: عندما يسجل كل المستخدمين نفس جهة الإصدار فإنه سيكون هناك ثقة مشتركة في جهة الإصدار تلك، ويمكن أن توضع كل الشهادات في دليل واحد متاح لجميع المستخدمين أو يتم تبادل الشهادات من مستخدم لآخر مباشرة كما هو في نظام الخصوصية فائقة الحسن PGP (Pretty Good Privacy).

شكل 16.3 نموذج الثقة المباشرة



1. نموذج الثقة الهرمية: يستعمل هذا النموذج لمجتمع كبير من المستخدمين، إذ يصبح أكثر عمليّاً أن يكون هناك عدد من جهات الإصدار كل واحدة منها توفر بشكل آمن مفتاحها العام لجملة من المستخدمين (انظر إلى الشكل 17.3).

شكل 17.3 نموذج الثقة الهرمية



يتم صناعة شجرة الثقة Trust Tree مكونة بشكل هرمي من الجذر ومن بقية جهات الإصدار والمصادقة، تستمد الثقة من عدد من الشهادات التي يصدرها الجذر. هذه الشهادات يمكن لها أن تصادق على الشهادات ذاتها، أو على الشهادات التي تصادق على شهادات دونها في التسلسل الهرمي. الشهادات الموجودة في طرف الشجرة يقع التأكد من صحتها بالتتبع الخلفي للجهة المصادقة عليها مباشرة، فالجهة التي أعلى منها في الشجرة إلى أن نجد شهادة مصادقًا عليها من الجذر مباشرة.

2. الثقة الهرمية والشهادات المتقاطعة:

لنفترض أن زيدًا (A) وعبيدًا (B) حصلا على شهادات X1 وX2 من جهات المصادقة .CAs لو أن زيدًا (A) لا يعلم بشكل آمن المفتاح العام لـX2

فإنه لا يمكن له أن يتحقق من صحة شهادة عبيد (B). لكن لو أن جهات المصادقة تبادلت فيما بينها مفاتيحها العامة، فإن زيدا سيحصل من خلال الدليل على شهادة X2 موقعة من طرف X1. فيمكن حينئذ لزيد أن يحصل على المفتاح العام ل X2. ثم يحصل زيد مرة ثانية على شهادة عبيد الموقعة من طرف X2 التي أصبح زيد الآن قادرًا على التحقق من صحتها من خلال النسخة الموثوقة من المفتاح العام لX2 التي تحصل عليها آنفًا. في معيار X.509 سلسلة الشهادات هذه يعرب عنها بما يلي:

$$\boldsymbol{X1} \ll \boldsymbol{X2} \gg \boldsymbol{X2} \ll \boldsymbol{B} \gg$$

ويحصل عبيد على مفتاح زيد بالسلسلة العكسية التالية:

$$\boldsymbol{X2} \ll \boldsymbol{X1} \gg \boldsymbol{X1} \ll \boldsymbol{A} \gg$$

وبشكل عام تكون السلسلة المتكونة من n عنصر كالآتي:

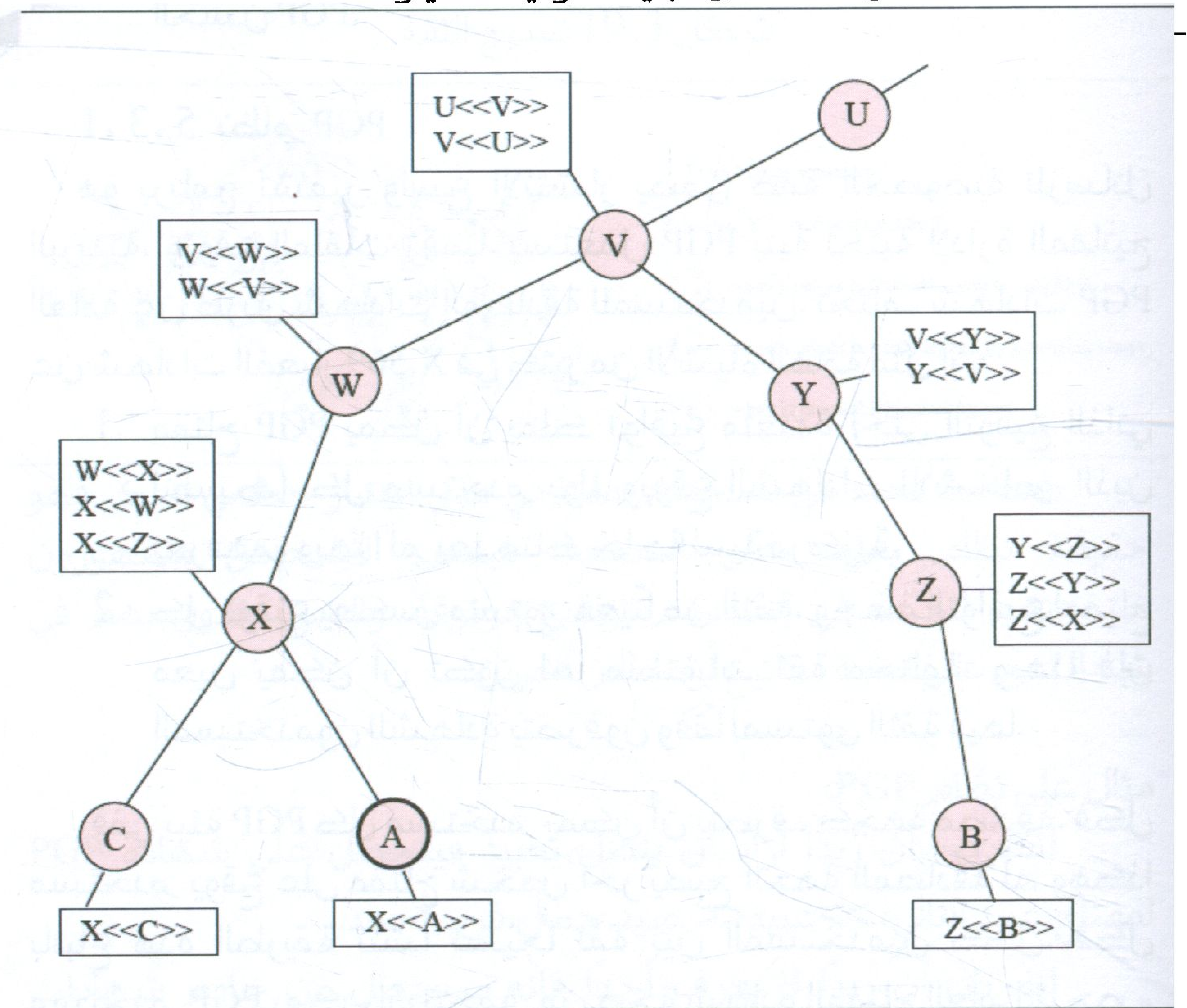
$$\boldsymbol{X1} \ll \boldsymbol{X2} \gg \boldsymbol{X2} \ll \boldsymbol{X3} \gg \ldots \boldsymbol{Xn} \ll \boldsymbol{B}$$

حيث إن كل زوج (X_i, X_{i+1})من الجهات المصادقة CAs في السلسلة صنعت كل واحدة منهما للأخرى شهادة مصادقة يحتفظ بها في الدليل.

يقترح معيار X.509 أن تكون جهات المصادقة CAs منظمة في شكل هرمي (انظر إلى الشكل الموالي). حيث إن العقد المرتبطة ببعض تمثل العلاقات بين جهات المصادقة والصناديق المربعة تمثل الشهادات التي تحويها الأدلة في كل جهة مصادقة. ونتحدث حينئذ عن الشهادات الأمامية والشهادات العكسية، فالأمامية هي الشهادات لجهة ما X والمولدة من طرف جهات مصادقة أخرى. والشهادات العكسية هي تلك المولدة من طرف X، والتي ولدت من طرف جهات الإصدار الأخرى.

مثال للبنية الهرمية للمعيار X.509:

شكل 3. 18مثال للبنية الهرمية للمعيار X.509



يمكن لزيد (المستخدم (A) أن يحصل على الشهادات التالية من الدليل لإنشاء مسار اتصال نحو عبيد (المستخدم (B).

X ≪ W ≫ W ≪ V ≫ V ≪ Y ≫ Y ≪ Z ≫ Z ≪ B ≫

وهكذا يحصل زيد على نسخة موثوقة من المفتاح العام لعبيد، وبدوره يمكن لعبيد الحصول على نسخة موثوقة من المفتاح العام لزيد عن طريق إنشاء مسار الاتصال التالي:

$$Z << Y >> Y << V >> V << W >> W << X >> X << A >>$$

3. **نسيج الثقة:** هذا النموذج يدمج بين النموذجين السابقين، وهما المباشر والهرمي، فالشهادة تكون موثوقة إما عن طريق مباشر، أو

عن طريق سلسلة من الجهات الموثوقة التي تنتهي بالجذر. من أمثلة الأنظمة التي تعتمد على نسيج الثقة نظام الخصوصية الفائقة الحسن PGP.

1. 3. 5 نظام PGP

هو برنامج تشفير واسع الانتشار يضمن صفة الخصوصية للرسائل البريدية، ويوقع الملفات رقميًا. يستعمل PGP بنية تحتية لإدارة المفاتيح العامة عن طريق شهادات المصادقة للمستخدمين. تختلف شهادات PGP عن شهادات المعيار X.509 في كثير من الأشياء الهامة مثل أن:

1. مفتاح PGP يمكن أن يملك تواقيع متعددة (حتى التوقيع الذاتي لنفسك). كل مستخدم يولد ويوقع الشهادات للأشخاص الذين يعرفهم، وبهذا لم يعد هناك حاجة لبنية مركزية.
2. كل توقيع يعكس مستوى معينًا من الثقة، وجملة التواقيع لمفتاح معين يمكن أن تكون لها مستويات ثقة مختلفة؛ وبهذا فإن المستخدمون للشهادة يتصرفون وفقًا لمستوى الثقة فيها.

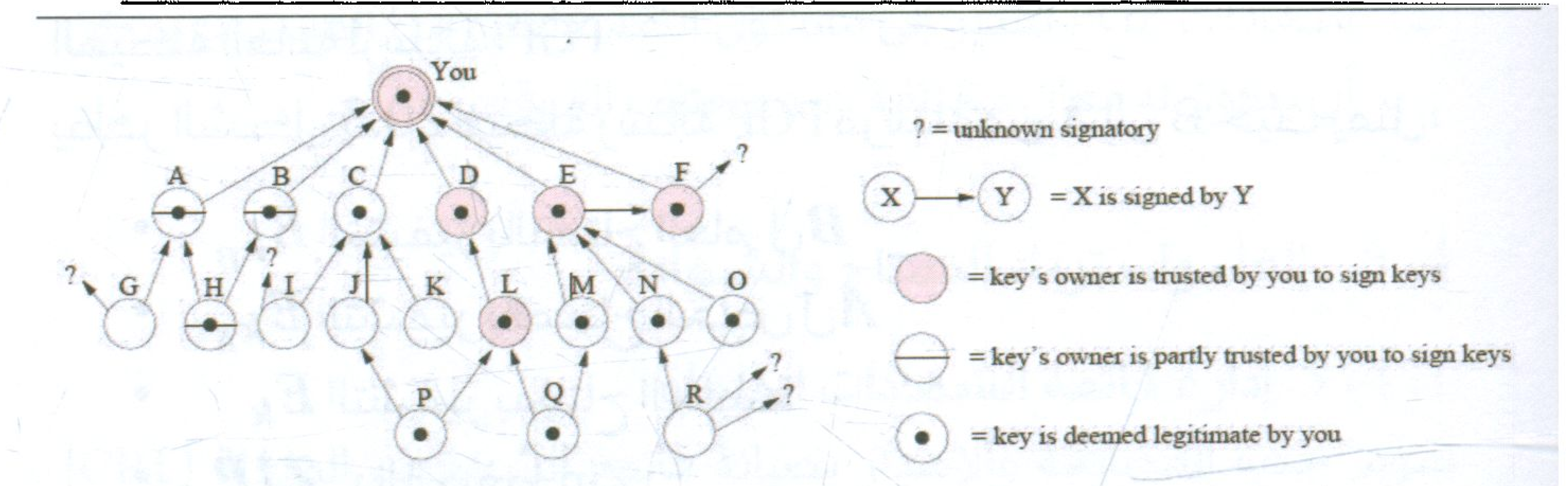
ففي بيئة PGP كل مستخدم يمكن أن يتصرف كجهة مصادقة. فكل مستخدم يوقع على مفتاح شخص آخر يصبح الجهة المصادقة له، وهكذا باتباع هذه الطريقة ننشأ نسيجًا ثقة بين المستخدمين ككل. فكل مستخدم PGP يمكن أن يتحقق من صحة شهادة المفتاح العام لشخص، ولكن هذه الشهادة لا تكون صحيحة للمستخدم الآخر؛ إلا إذا كان يعتبر المتحقق ثقة عنده، فهو لا يثق في حكمه على الشهادة إلا إذا كان ثقة عنده.

لكل مستخدم مؤشرات على المفاتيح التي عنده تحدد:

4. ما إذا يعتبر المستخدم مفتاحا معينًا صحيحًا عنده.
5. مستوى الثقة التي يمنحها للمفتاح التي تعكس مدى ثقة المستخدم في مصداقية مالك ذلك المفتاح في مصادقته على المفاتيح الأخرى.

يوضح الشكل 19.3 هذه المؤشرات فيمكن أن يكون المفتاح في ذاته صحيحًا عندك، ويمكن أن تكون ثقتك في صاحب المفتاح جزئية أو تامة.

شكل 19.3 نسيج الثقة



فكأنك تحدد في نسختك لمفتاحي مدى اعتبار حكمي لديك. فهو حقيقة نظام يعتمد على السمعة: فبعض الأشخاص مشهورون بسمعتهم الجيدة في إسنادهم تواقيع موثوقة، والناس يثقون بهم في شهادتهم على صحة مفاتيح أشخاص آخرين.

مثال على نظام PGP:

لنفترض أن زيدا أراد أن يتصل بعبيد فيتحصل على شهادة PGP لمفتاح عبيد التالي: >>عبيد<< عبيد،جمال،فريد،أحمد.

لنفرض أن زيدًا لا يعرف أحدًا فإنه سيحصل من خادم شهادات شهادةPGP الخاصة بجمال ولتكن التالية: >> جمال<< جمال، أيمن، هاشم.

لنفرض أيضًا أن زيدًا لا يعرف هاشمًا بشكل واضح؛ فلكي يتحقق من شهادة جمال يطلب الحصول على شهادة لهاشم ولتكن التالية: >>هاشم<< هاشم،أحمد. لاحظ زيد أن مستوى الثقة في توقيع هاشم ليس بذاك، فقرر أن يبحث عن شيء آخر يدعمه. لنفرض أن زيدًا تحصل على شهادة أمجد، ولتكن التالية: >>أحمد<< أحمد،جابر ومباشرة عرفت أن جابرًا هو ابن عمه الذي يثق فيه مما جعله يثق بشهادة أحمد التي يستند إليها في اعتبار أن شهادة عبيد موثوقة، وبالتالي فإن المفتاح التي تحويه هو

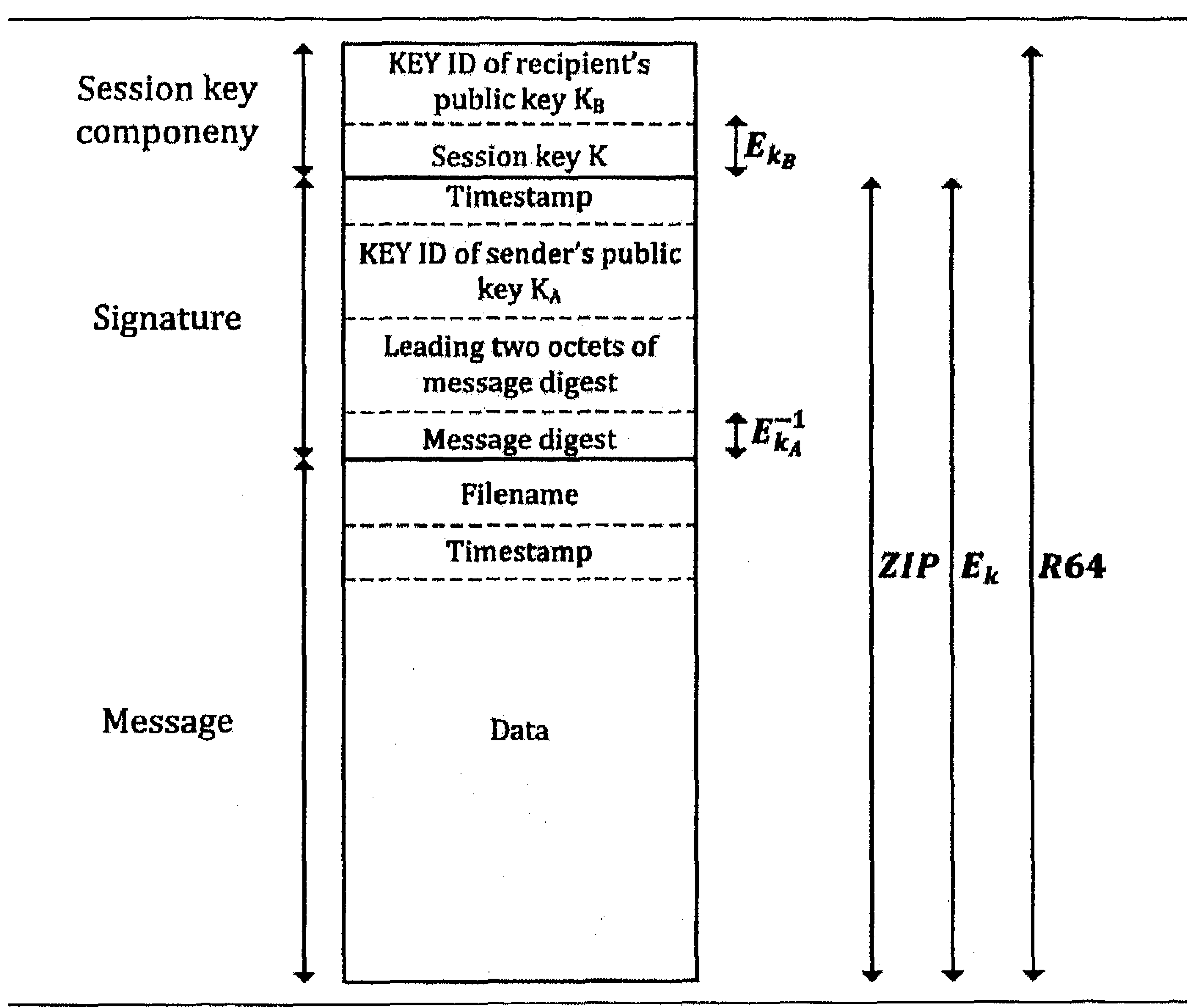
مفتاح عبيد حقيقة. في هذا المثال استعمل زيد سلسلتين، وهما: سلسلة <<عبيد>> جمال <<جمال>> هاشم <<هاشم>> هاشم وسلسلة<<عبيد>> أحمد <<أحمد>> جابر.

الهيكلة العامة لرسالة PGP:

يظهر الشكل 20.3 هيكلة رسالة PGP مرسلة من A إلى B حيث يمثل:

- E_{k_B} التشفير بالمفتاح العام لـB
- E_{k_A} التشفير بالمفتاح الخاص لـA
- E_k التشفير بمفتاح الجلسة
- ZIP دالة ضغط Zip
- $R64$ دالة ضغط Radix-64

شكل 20.3 هيكلة رسالة PGP



وللتمييز بين شهادات X.509 وPGP فإن

1. شهادات X.509 تحوي عنصر ثقة، ولكن الثقة غير مشار إليها في الشهادة.
2. شهادات PGP تشير إلى مستوى الثقة، ولكن هذا المستوى يمكن أن يكون له معانٍ مختلفة عند مختلف الموقعين.

5 .4 إلغاء واسترداد المفتاح والشهادة

5 .4 .1 إدارة قائمة الشهادات الملغاة

تقوم جهة المصادقة والإصدار بصيانة قائمة الشهادات الملغاة (CRL) والتي لم تنتهِ مدة صلاحيتها التي أصدرتها، ثم وضعتها في الدليل لإتاحتها للمستخدمين. ثم إما أن يتحقق المستخدمون أنفسهم من هذه القائمةن أو يستعملون خدمة التحقق التي تجمع وتتحقق بشكل مركزي من قائمة الشهادات الملغاة. يقع إلغاء الشهادة لعدة مرّات متى ما وقع المساس بالمفتاح الخاص لمستخدم، أو المساس بشهادته المصادق عليها، أو أن المستخدم لم يعد من ضمن الذين تشهد له جهة الإصدار هذه. في معيار X.509 كل شهادة لها مدة صلاحية محددة كما سبق الإشارة إلى ذلك، وعادة ما تصدر للمستخدم شهادة جديدة قبل أن تنتهي مدة صلاحية الشهادة القديمة.

تحتوي قائمة الشهادات الملغاة أساسًا على العناصر التالية (انظر إلى الشكل الموالي)

1. اسم مصدر الشهادة.
2. تاريخ إصدار قائمة الشهادات الملغاة.
3. التاريخ المقبل لإصدار القائمة.
4. مدخل لكل شهادة ملغاة.

شكل 3. 21 عناصر الشهادة الملغاة.

Signature algorithm identifier: Algorithm / Parameters
Issuer name
This update date
Next update
Revoked certificate: User certification serial # / Revocation date
•
•
•
Revoked certificate: User certification serial # / Revocation date
Signature: Algorithms / Parameters / Encrypted

2. 4. 5 استرداد المفتاح

السؤال هنا: كيف يمكن لزيد أن يسترد مفتاحه لو فقده، أو أن الأشخاص الذين يعرفونه ليس بقدرتهم أو لا يريدون الكشف له عنه؟ هذه المسألة خطيرة جدًا، خصوصًا لو ربط المفتاح بدور معين للشخص، كأن يقال صحب المفتاح الفلاني له أحقية الفعل الفلاني. هناك ثلاث خيارات في التعامل مع هذه المسألة: إما أن يكون المفتاح أو نظام التشفير ضعيفًا، أو أن يتم وضع نسخة من المفتاح في مكان ما، أو يوجد نظام فتح مفاتيح KeyEscrow بحيث يمكن لكيان آخر أن يسترد المفتاح من جديد. كما يكون في المعاملات التجارية، حيث يتم الاسترجاع من النسخ الاحتياطية، أو لإنفاذ القانون من طرف سلطة ما لو حصل تجاوز ما.

عادة ما يكون لكل مستخدم زوجين من المفاتيح، الزوج الأول يستعمل في التشفير، والزوج الثاني في التوقيع الالكتروني. فمفتاح فك التشفير يكون منه نسخة احتياطية في جهة الإصدار CA وهكذا يمكن استرداده فيما لو فُقِدَ، وأيضًا يمكن فتحه بنظام فتح المفاتيح، أما مفتاح التوقيع الالكتروني وأيضًا مفتاح ضمن عدم الإنكار فوضع نسخة احتياطية في جهة المصادقة غير منصوح به، والأولى أن لا يخرج من حوزة المستخدم.

5 .5 التسمية والهوية

نقوم في عملية التأكد من الهوية بالربط بين كيان ما مع معرف هويته في الحاسب. كل نظام له طريقته الخاصة في تعريف هوية كيان ما، ولكن كل القرارات في مراقبة الدخول، والوصول للموارد تفترض أن هذا الربط بين الكيان وهويته صحيحًا. يتم تعريف الأشياء الداخلية للحاسب بتعيين اسم لكل شيء منها، وإن كان هذا الشيء على الشبكة فيتم تعيينه بتحديد معرف Uniform Resource Locator (URL) له.

1 .5 .5 الأسماء في بنية PKI

يربط PKI بين مفتاح زيد العام واسمه. ولكن ما ماهية الاسم؟

إن كنا في مجتمع محدود كقرية صغيرة مثلاً، حيث كل شخص يعرف الآخرين ولو بالشكل فقط فإن كل شخص سيملك اسمًا يعرف به الاسم. فلكل اسم صاحبه المعروف وإن كان هذا الشخص يمكن أن يملك أكثر من اسم. فإن استحالت هذه القرية مدينة فهذه الأسماء تفقد ارتباطها المباشر مع الشخص، حيث إننا لم نعد نعرف كل الناس في هذه المدينة، ثم إذا اتسعت المدينة أكثر فإننا لا نعرف إلا مجموعة محدودة من الناس فيها، والأسماء لم تعد فريدة، ومعرفة الشخص تتم عن طريق سياق الكلام. أما في الإنترنت فقد حل عنوان البريد الإلكتروني محل الاسم. وهذا العنوان – وإن كان فريدًا – فهو لا يعكس معرفتنا بالشخص، حتى وإن علمنا عنه كثيرًا من المعلومات كتاريخ الولادة، والعنوان البريدي، وغير ذلك. بل أكثر من ذلك فإن الشخص الواحد بإمكانه أن يكتسب أكثر من بريد الكتروني بمعلومات مختلفة. وكذلك يمكن لمجموعة من

الأشخاص المختلفين الاشتراك في بريد الكتروني واحد، هو بريد المجموعة.

لكي تحصل على تقابل وحيد بين الأشخاص وأسمائهم، تسعى الحكومات والمؤسسات لتعيين معرف فريدٍ لكل أحد، مثل أن نحدد الاسم وتاريخ الميلاد والعنوان واسم الأب والجد واسم الأم والجنسية. ولكن بعض هذه المعلومات يمكن أن تتغير عبر الزمن، وأيضًا بعض الناس يملك أكثر من جنسية، وبعضهم لا يملك أصلاً جنسية فيصعب إذن أن نحصل على تغطية شاملة بشكل أحادي لكل الأفراد ولكن نستطيع أن نحدد بعض السياسات والطرق في وضع معرف أو اسم لكيان م وأيضا في التعرف على كيان ما من خلال اسمه أو معرفه.

يستخدم المعيار X.509 اسمان في هيكل الشهادة وهما:

1. اسم الجهة المصدرة CA المعطى من X.509 والتي وقعت على الشهادة، ولو تم استعمال هذا الاسم من أكثر من كيان فإن هناك اسما فريدًا اختياريًا آخرَ.
2. اسم المستخدم A الذي سيتم ربطه بمفتاحه العام في الشهادة، ولو تم استعمال هذا الاسم من أكثر من كيان فإن هناك اسْمًا فريدًا اختياريًا آخر.

عندما تصادق الجهات المصدرة للشهادات فكأنما تصادق بذلك على هوية الشخص الذي أصدرت له الشهادة، فتحتاج هذه الجهات إلى سياسة تأكد من الهوية تحدد كيف تتعامل مع ادعاء كيان ما هوية ما، من حيث التأكد من صحة ادعائه، ومن حيث مصداقية إعطائه الشهادة. كما تحتاج إلى سياسة توصف خصائص الأشخاص الذين ستصدر لهم الجهة الشهادات، وكيفية اتخذ قرار إصدار شهادة لكيان ما من عدمه. أخيرًا يتعين على الجهات المصدرة أن تمنع التضاد بين الشهادات، ولكن كل من المعيار X.509 وPGP تلزم الصمت حول هذه المسألة.

6 - أمن البرتوكولات

تقدم معنا عدة حلول لعدة متطلبات أمنية، ولكن هذه الحلول تحل مشكلة جزئية من المسألة، وتطلب أشياءَ إضافيةً في الواقع العملي. فمثلا المفاتيح العامة يمكن أن توزع وتتاح للجميع، ولكن لا بد من التأكد من هوية الرسالة. وكذلك برتوكول ديفي هيلمن DH يقوم بتوليد مفاتيح سرية مشتركة، ولكن يتطلب أيضًا التأكد من الهوية. أما التوقيع الالكتروني فإنه يقوم بالتأكد من هوية الرسالة، ولكنه لا يضمن أن التوقيع وقع فعلا في الوقت المحدد في الرسالة(Timeliness of message). والهدف الذي نسعى إليه: هو كيف نؤلف بين هذه الخدمات بحيث نضمن خصائص أمنية متكاملة تحتاجها عدة خدمات شبكية؟ كالتجارة الإلكترونية، والحكومة الإلكترونية، وغيرها. فمثلا لو طلب زيد من عبيد أن يحول له مبلغًا ماليًا ما لحساب معين، فأجابه عبيد أنه سيقوم بهذا الآن فالسؤال هو: كيف تأكد عبيد أن مخاطبه هو زيد وأن هذه الرسالة وقع إرسالها فعلا من زيد قبل لحظات؟. هناك عدة برتوكولات عملية لحل هذه المسألة مثل IPSEC, SSH, PGP, SSL,SET, Kerberos سنعرض إليها في الفقرات المقبلة، ولكن في هذه الفقرة سنركز على الأفكار الأساسية التي تعتمد عليها هذه البرتوكولات.

6.1 تعريفات أساسية

تعرف برتوكولات الأمن على أنها برتوكولات تستعمل آليات وتقنيات التشفير لتحقيق الأهداف الأمنية. وهي في حد ذاتها برتوكولات صغيرة ا لحجم، ولكن تصميمها وفهمها ليس بديهيًا.

ليكن زيد وعبيد هما طرف الاتصال وسنرمز لزيد ب A لعبيد ب B .

ولنرمز لمفتاح التوقيع الالكتروني والتحقق منه ب $\boldsymbol{K^{-1}}$و K ولعملية التشفير للرسالة M ب $\{M\}_k$ والتوقيع ب$\{\boldsymbol{M}\}_{\boldsymbol{k}^{-1}}$. والتشفير بالمفتاح العام لزيد $\{M\}_{k_A}$ وإمضاؤه $\{\boldsymbol{M}\}_{k_A^{-1}}$. والتشفير التماثلي بالمفتاح المشترك بين زيد وعبيد ب$\{M\}_{k_{AB}}$. ولتكن N_A الرقم العشوائي Nonce الذي سنستعمله كعنصر التحدي في برتوكولات التأكد من الهوية. ليكن T الطابع زمني

timestamp الذي سيستعمل مثلا في تحديد صلاحية المفتاح، ولترمز $\{C, M_2\}$ للرسالة المؤلفة من M_1 وM_2.
نمثل الاتصال بين طرفين بما يلي:

$$A \rightarrow B : \{A, T_A, K_{AB}\}_{K_B}$$

حيث A و B هما طرف الاتصال، ويمكن أن يُعوضَا بأي طرفين آخرين، ونفترض أن الاتصال غير متزامن وكذلك $A \rightarrow B$ ليس جزءا من الرسالة. فيكون تعريف البروتوكول هو: توصيف مجموع ما يتبادله طرفا الاتصال بينهما بشكل تسلسلي معين.
هناك فرضيات وأهداف وأطروحات في أي برتوكول. فالفرضيات هي:

- كل طرف يعرف مفتاحه العام والخاص، ويعرف المفتاح العام لكل طرف يريد أن يتصل به.
- كل طرف قادر على توليد رقم تَحَدٍ عَشوائي متى ما أراد ذلك

والأهداف هي:

- التأكد من هوية الرسالة والمرسل.
- التأكد من أن الرسالة حديثة وليست قديمة.
- ضمان سرية المفاتيح المولدة.

أما الأطروحات فهي أن

- كل برتوكول بدون أهداف وفرضيات واضحة لا قيمة له.
- كل برتوكول بدون إثبات لصحته يحتمل أن يكون سيئًا.

6.2 أنواع الهجومات على البروتوكولات

6.2.1 تعريف المهاجم

بداية لا بد من تعريف المهاجم، فالتعريف المعياري هو الكيان الذي يعرف البرتوكول، ولكن لا يستطيع كسر الشفرة. يمكن أن يكون خاملاً، ولكن بإمكانه التنصت على كل الاتصالات كما أنه يمكن أن يكون نشطًا، فيقوم باعتراض وتوليد الرسائل. يمكن أن يكون المهاجم أحد طرفي الاتصال غير المشكوك فيهما أصلا كما يقول الشافعي رحمه الله:

احْـــــذَرْ عَـــــدُوَّكَ مَـــــرَّةً وَاحْـــذَرْ صَـــدِيْقَكَ أَلْـــفَ مَـــرَّهْ
فَلَرُبَّمَـــا انْقَلَـــبَ الصَّدِيْـــــقُ فَكَـــانَ أَعْلَـــمَ بِالْمَضَـــرَّهْ

ويعتبر المهاجم نَشِطًا حسب نموذج Dolev-Yao إذا أمكنه أن يقوم بـ:

1. اعتراض وقراءة كل الرسائل.
2. تقسيم الرسالة لأجزائها، ولكن التشفير هنا سيعيقه من هذا الفعل.
3. توليد وإنشاء رسائل جديدة مختلفة.
4. إرسال الرسائل في أي وقت ومتى ما شاء.

وبشكل عام فإنه بقدر افتراضنا أعلى الإمكانيات للمهاجم بقدر ما يكون البرتوكول أكثر قوة وصحة وقابلية؛ لأن ينفذ في مختلف البيئات.

2. 2. 6 أهم الهجومات على البروتوكولات

بداية نذكر أهم الهجومات التي تقع على البرتوكولات وهي:

1. هجوم إعادة إرسال الرسائل المرسلة في السابق (Replay Attack)، أو ما يعبر عنه أيضًا بـ(Freshness Attack)، وهو أن يستعمل الدخيل رسائل أرسلت سابقًا في تنفيذ هجومه على أحد طرفي الاتصال.
2. هجوم الدخيل الذي في الوسط بين طرفي الاتصال (Man-in-the-middle Attack) أو ما يعبر عنه أيضًا بهجوم الجلسات الموازية(Parallel sessions Attack) وهو أن يكون الدخيل في الوسط ويتقمص شخصية كل طرف للطرف الثاني، ويقوم بالاطلاع، وإنشاء الرسائل دون شعور الطرفين به.
3. هجوم تقمص الهوية (Masquerading Attack) ويكون بأن يزيف الدخيل عنوان مصدر الرسالة في طرد الانترنت أو أن يقنع الآخرين أن مفتاحه هو المفتاح العام لزيد.
4. هجوم تغيير نوع الحقول (Type flaw Attack): وهو أن يقوم الدخيل بتغيير نوع حقول الرسالة فمثلا: عِوَضَ أن يرسل رقمًا عشوائيًا كَتَحَدٍ يرسل مكانه اسمًا. أي مجموعة أحرف لا رقم.
5. هجوم الارتداد (Reflection Attack) وهو أن يقوم الدخيل بإرجاع الرسائل المرسلة لمرسلها.

6 .3 أمثلة من الهجومات على البروتوكولات

1. 3. 6 برتوكول NSPK (Needham-Schroeder Public Key)

يتمثل بروتوكول NSPK المطور في 1978 فيما يلي :

$$1.\ A \rightarrow S: A, B$$

$$2.\ S \rightarrow A: \{K_B, B\}_{K_S^{-1}}$$

$$3.\ A \rightarrow B: \{A, N_A\}_{K_B}$$

$$4.\ B \rightarrow S: B, A$$

$$5.\ S \rightarrow B: \{K_A, A\}_{K_S^{-1}}$$

$$6.\ B \rightarrow A: \{N_A, N_B\}_{K_A}$$

$$7.\ A \rightarrow B: \{N_B\}_{K_B}$$

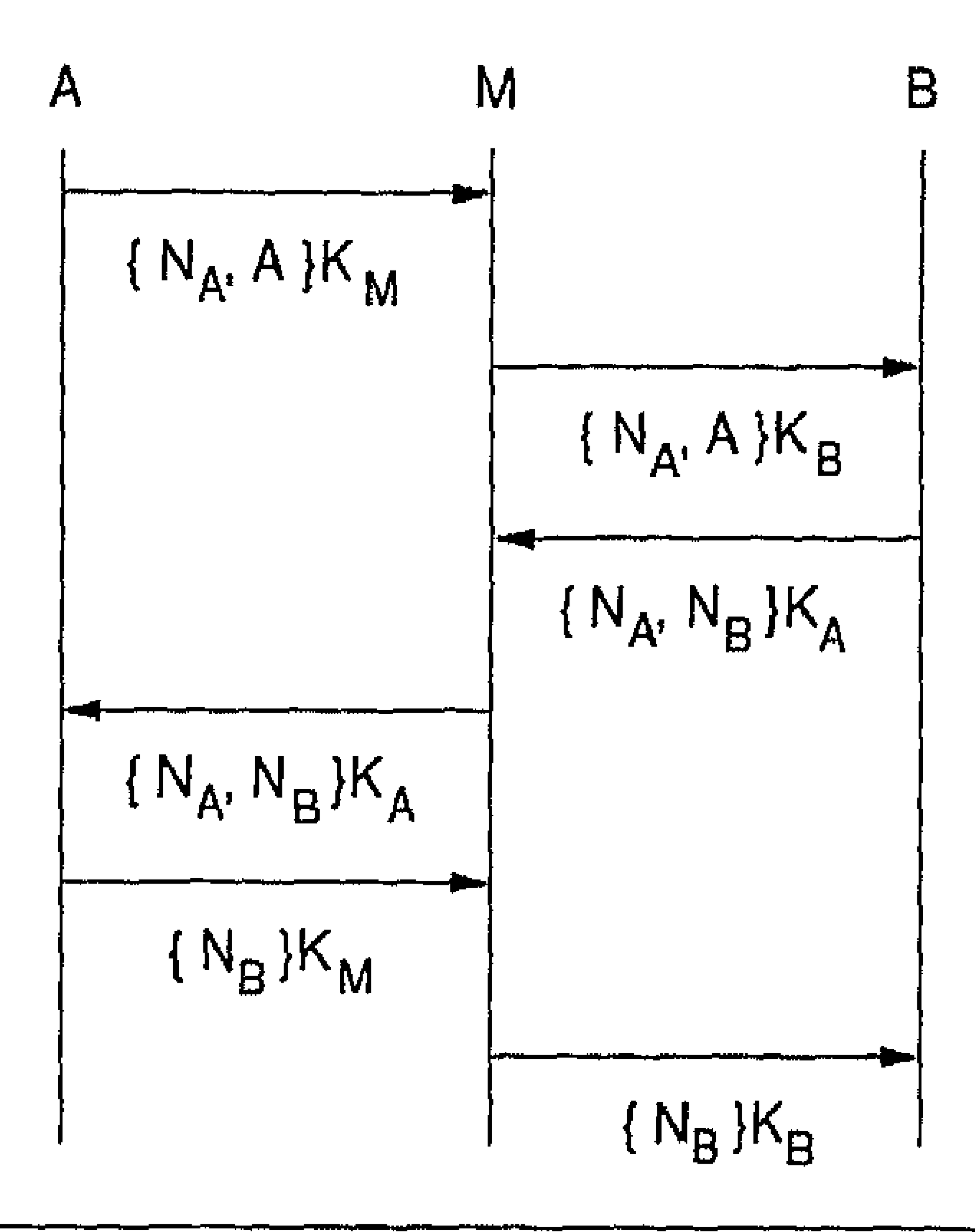
1. يرسل زيد إلى S وهي الجهة التي تمتلك توزيع المفاتيح العامة رغبته في الاتصال بعبيد.
2. ترسل S مفتاح عبيد واسمه موقعين بمفتاحها الخاص، فيفتح زيد الرسالة الموقعة ويستخلص المفتاح العام لعبيد.
3. يرسل زيد اسمه ورقم تَحَدٍ مشفرين بالمفتاح العام لعبيد.
4. يرسل عبيد إلى S وهي الجهة التي تمتلك توزيع المفاتيح العامة رغبته في الاتصال بزيد.
5. ترسل S مفتاح زيد واسمه موقعين بمفتاحها الخاص فيفتح عبيد الرسالة الموقعة ويستخلص المفتاح العام لزيد.
6. يفتح عبيد الرسالة بمفتاحه الخاص، ويستخلص رقم التحدي المرسل من زيد ثم يؤلف معه رقم تحدٍ منه لزيد، ويشفر الجميع بالمفتاح العام لزيد. يفتح زيد الرسالة ويستخلص رقم تحديه ويقارنه مع رقمه الأصلي؛ فإن تساويا فالمرسل هو عبيد لأنه هو الوحيد القادر على فتح رسالة مشفرة بمفتاحه العام لامتلاكه المفتاح الخاص.

7. يستخلص زيد أيضًا رقم تحدٍ عبيد الذي يشفره بالمفتاح العام لعبيد، ويرسله له. يفتح عبيد الرسالة بمفتاحه الخاص، ويستخلص الرقم المرسل من زيد فإن تساوى مع رقم تحديه فالمرسل هو زيد، لأنه هو الوحيد القادر على فتح رسالة مشفرة بمفتاحه العام لامتلاكه المفتاح الخاص.
8. تمت عملية التأكد من الهوية من الطرفين

في عام (1996) نشر Lowe إمكانية تعرض NSPK لهجوم الدخيل الذي في الوسط (Man-in-the-middle attack) وذلك كالآتي:

1. يفتتح زيد (A) جلسة NSPK مع شخص يعرفه وهو (M). فيقوم M بهجوم الدخيل الذي في الوسط مع عبيد (B). إذ يفك شفرة الرسالة التي وصلته من زيد ويرسلها مشفرة بمفتاح عبيد.
2. يظن عبيد أن الرسالة أرسلت من زيد، فيفكها ويرسل تحدي زيد مع تحديه مشفرين بمفتاح زيد.
3. يقوم الدخيل بتمرير الرسالة كما هي لزيد فيظن زيد أن الرسالة قد أرسلها M، لأنه لا يمكنه معرفة أن رقم التحدي هو من عبيد وليس من الدخيل، إذ هو مجرد رقم.
4. يقوم زيد بفك الشفرة، وإرسال رقم التحدي للدخيل، وهنا تعرف الدخيل على شيء في الأصل لا يمكنه التعرف عليه، إذ لا يمكن أن يعرفه إلا زيد وعبيد.
5. يقوم الدخيل بتشفير التحدي بمفتاح عبيد العام، ولما تصل الرسالة لعبيد سيتحقق أن زيدًا هو الذي يخاطبه، لأن رقم التحدي المستقبل مساوٍ لرقمه الذي أرسله، وهكذا نجح الدخيل في إيهام عبيد أنه زيد. (انظر الشكل 22.3)

شكل 3. 22 هجوم الدخيل الذي بالوسط على NSPK



يجدر أن نشير إلى أن برتوكول NSPK استعمل لمدة 18 سنة دون أن تكتشف هذا الهجوم الذي يمس في آن واحد السلامة والتأكد من الهوية.

اقترح Lowe الحل التالي لهذا الهجوم:

شكل 3. 23 حل Lowe

A
M
B
$\{ N_A, A \}K_M$
$\{ N_A, A \}K_B$
$\{ N_A, N_B, B \}K_A$
$\{ N_A, N_B, M \}K_A$

$$1.\ A \rightarrow S: A, B$$

$$2.\ S \rightarrow A: \{K_B, B\}_{K_S^{-1}}$$

$$3.\ A \rightarrow B: \{A, N_A\}_{K_B}$$

$$4.\ B \rightarrow S: B, A$$

$$5.\ S \rightarrow B: \{K_A, A\}_{K_S^{-1}}$$

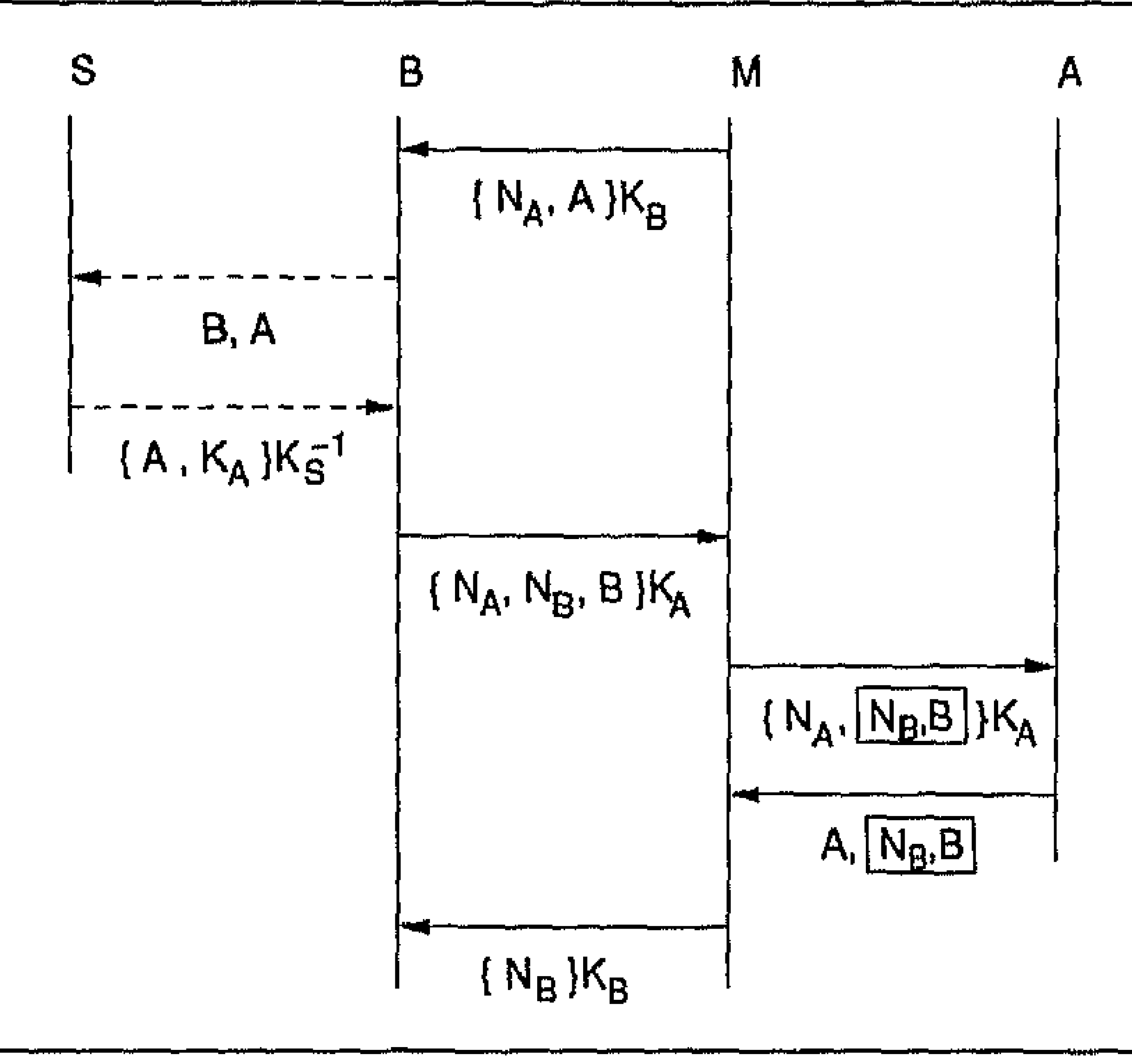
$$6.\ B \rightarrow A: \{N_A, N_B, B\}_{K_A}$$

$$7.\ A \rightarrow B: \{N_B\}_{K_B}$$

وهو أن يضيف عبيد اسمه عندما يرسل تحديه وتحدي الطرف الثاني، ليعلم زيد أن الرسالة ولدها عبيد وليس الدخيل الذي من المفترض أن يولد هو الرسالة، ليتوافق مع البرتوكول فيمتنع على إجابة الدخيل ويكتشف الهجوم. أما الدخيل فإنه لا يستطيع القيام بهجومه إلا إذا استطاع أن يدرج اسمه في الرسالة التي أرسلها عبيد، وهذا يتطلب منه معرفة المفتاح الخاص لزيد، وبينه وبين ذلك خرط القتاد.

ولكن هذا الحل – وإن كان قد صد هجوم الدخيل الذي في الوسط – إلا أن Meadows في سنة (1996) أثبت إمكانية هجوم تغيير نوع الحقول على الحل المقترح، إذ لما يرسل عبيد رسالته التي تحوي اسمه، ورقم تحديه، ورقم تحدي زيد يقوم الدخيل بفتح جلسة جديدة من برتوكول NSPK مع زيد مرسلا إليه نفس الرسالة، فيجيبه زيد مرسلا إليه رقمَ تحديه الذي ما هو إلا الحقلان المدمجان لرقم تحدي عبيد السابق مع اسمه، فيستخلص الدخيل رقم التحدي ويرسله إلى عبيد مشفرا بمفتاحه العام، وهكذا ينجح في إتمام هجومه (انظر الشكل 24.3)

شكل 3.24 هجوم Meadows



2. 3. 6 بروتوكول Otway-Rees

يعمل هذا البرتوكول في خادم توزيع مفاتيح التشفير التماثلي، حيث إن المفتاح المشترك K_{AS} بين الخادم S وA معلوم، وكذلك المفتاح K_{BS} مع B وحيث يكون I رقمًا طبيعيًا يحدد برتوكولا معينًا:

$$\mathbf{1.}\, \boldsymbol{A \to B: I, A, B, \{N_A, I, A, B\}K_{AS}}$$
$$\mathbf{2.}\, \boldsymbol{B \to S: I, A, B, \{N_A, I, A, B\}K_{AS}, \{N_B, I, A, B\}K_{BS}}$$
$$\mathbf{3.}\, \boldsymbol{S \to B: I, \{N_A, K_{AB}\}K_{AS}, \{N_B, K_{AB}\}K_{BS}}$$
$$\mathbf{4.}\, \boldsymbol{B \to A: I, \{N_A, K_{AB}\}K_{AS}}$$

يخضع هذا البروتوكول لهجوم تغيير نوع الحقول Type Flaw attack إذ لو افترضنا أن $I = 32bits, A = 16\ bits, B = 16bits, K_{AB} = 64\ bits$ فإن طول $|\{K_{AB}\}| = |\{I, A, B\}|$ فيمكن أن يقوم الدخيل M بالهجومين التاليين:

1. يلغي الخطوة الثانية والثالثة من البرتوكول، ويرد الرسالة للمرسل الأول مباشرة

$$\mathbf{1.}\, \boldsymbol{A \to M: I, A, B, \{N_A, I, A, B\}K_{AS}}$$
$$\mathbf{4.}\, \boldsymbol{M \to A: I, \{N_A, I, A, B\}K_{AS}}$$

يتأكد A من رقم تحديه N_A ويقبل مجموع I, A, B كمفتاح سري مشترك للجلسة مع B .

2. يلعب الدخيل دور الخادم في الخطوتين الثانية والثالثة برد الجزء المشفر في الخطوة الثانية لـB

$$\mathbf{1.}\, A \to B: I, A, B, \{N_A, I, A, B\}K_{AS}$$
$$\mathbf{2.}\, B \to M: I, A, B, \{N_A, I, A, B\}K_{AS}, \{N_B, I, A, B\}K_{BS}$$
$$\mathbf{3.}\, M \to B: I, \{N_A, I, A, B\}K_{AS}, \{N_B, I, A, B\}K_{BS}$$
$$\mathbf{4.}\, B \to A: I, \{N_A, I, A, B\}K_{AS}$$

وبهذا يقبل طرفا الاتصال المفتاح المشترك وهو مجموع I, A, B وبهذا يستطيع الدخيل الاطلاع الكامل على الجلسة بينهما لأن I, A, B معلومة لديه في الخطوة الثانية من البرتوكول.

3. 3. 6 بروتوكول Andrew Secure RPC

يهدف هذا البرتوكول إلى تبادل مفاتيح جلسة تكون سريةً، ومشتركةً وموثوقةً وحديثةً بين طرفين يملكان مفتاحًا للتشفير التماثلي بينهما.

$$\mathbf{1.}\, A \to B: A, \{N_A\}_{K_{AB}}$$
$$\mathbf{2.}\, B \to A: \{N_A + 1, N_B\}_{K_{AB}}$$
$$\mathbf{3.}\, A \to B: \{N_B + 1\}_{K_{AB}}$$
$$\mathbf{4.}\, B \to A: \{K'_{AB}, N'_B\}_{K_{AB}}$$

حيث إن K'_{AB} هو المفتاح المشترك لهذه الجلسة وN'_B عنصر التحدي للجلسة الموالية. فلو فرضنا أن طول المفتاح بنفس طول أرقام التحدي فإن الدخيل M بإمكانه أن يقوم بالهجوم التالي، وهو: أن يحتفظ بالبيانات المرسلة في الخطوة الثانية أي الرسالة الثانية ويعترض الرسالة الثالثة، ويقوم بإرسال الرسالة الثانية على أنها الرسالة الرابعة:

$$\mathbf{1.}\, A \to B: A, \{N_A\}_{K_{AB}}$$
$$\mathbf{2.}\, B \to A: \{N_A + 1, N_B\}_{K_{AB}}$$
$$\mathbf{3.}\, A \to M: \{N_B + 1\}_{K_{AB}}$$
$$\mathbf{4.}\, M \to A: \{N_A + 1, N_B\}_{K_{AB}}$$

وبهذا سيقبل A رقم تحديه مضافًا إليه رقم واحد $N_A + 1$ كمفتاح سري مشترك لهذه الجلسة. ورغم نجاح الدخيل في هذا الهجوم إلا أنه لا يمكن له الاطلاع على الرسائل المرسلة بين طرفي الاتصال، لأنه لا يعرف قيمة N_A.

4. 3. 6 بروتوكول Denning and Sacco

يهدف هذا البرتوكول إلى الاشتراك في مفتاح جلسة للتشفير التماثلي بالاعتماد على الشهادات.

$$\mathbf{A \rightarrow S: A, B}$$
$$\mathbf{S \rightarrow A: C_A, C_B}$$
$$\mathbf{A \rightarrow B: C_A, C_B, \left\{\{T_A, K_{AB}\}_{K_A^{-1}}\right\}_{K_B}}$$

المفتاح K_{AB} هو المفتاح المشترك A وB الذي تأكد من كونه مرسل من A ، لأنه وقع الرسالة بمفتاحه الخاص، والذي تم فتحه بالمفتاح العام من خلال استخلاصه من الشهادة C_A. كما يعلم B و أن الرسالة مرسلة إليه لأنّها مشفرة بمفتاحه العام. ويستعمل الطابع الزمني T_A لتحديد مدة صلاحية المفتاح. يمكن أن يقوم الدخيل بهجوم الدخيل الذي في الوسط وذلك بفعل الآتي:

$$A \rightarrow M: C_A, C_M, \left\{\{T_A, K_{AM}\}_{K_A^{-1}}\right\}_{K_M}$$
$$M \rightarrow B: C_A, C_B, \left\{\{T_A, K_{AM}\}_{K_A^{-1}}\right\}_{K_B}$$

فيعتقد B أن الرسالة أرسلت من طرف A ، وأن المفتاح المشترك مع A هو K_{AM} وهكذا أمكن للدخيل أن يتقمص شخصية A لـ B وأيضًا الاطلاع على كل الرسائل المتبادلة بينهما، لأنه يملك المفتاح المشترك K_{AM}. لمنع مثل هذا الهجوم لا بد من وضع اسم طرفي الاتصال في الرسالة الأخيرة المتبادلة بين A وB فيصبح البرتوكول كما يلي:

$$A \rightarrow S: A, B$$
$$S \rightarrow A: C_A, C_B$$
$$A \rightarrow B: C_A, C_B, \left\{\{A, B, T_A, K_{AB}\}_{K_A^{-1}}\right\}_{K_B}$$

5. 3. 6 مزالق بعض البروتوكولات المستعملة في التأكد من الهوية

- البرتوكول الأول:

$$A \to B: A$$
$$B \to A: N_B$$
$$A \to B: \{N_B\}_{K_{AB}}$$

- البرتوكول الثاني:

$$A \to B: A$$
$$B \to A: N_B$$
$$A \to B: MD\{N_B | K_{AB}\}$$

الهجوم الذي يمكن أن يحدث على كلا البرتوكولين السابقين، هو أن الدخيل يمكن أن يقوم بمحاولة كسر الشفرة، أو البصمة من خلال تنصته على الاتصال ومعرفته ب N_B و $\{N_B\}_{K_{AB}}$ أو $MD\{N_B | K_{AB}\}$.

- البرتوكول الثالث:

$$A \to B: A$$
$$B \to A: \{N_B\}_{K_{AB}}$$
$$A \to B: N_B$$

يستطيع الدخيل الحصول على النصوص المشفرة بدون التنصت على الاتصال.

- البرتوكول الرابع:

$$A \to B: A, \{T_A\}_{K_{AB}}$$

- البرتوكول الخامس:

$$A \to B: A, T_A, MD\{K_{AB} | T_A\}$$

كل من البرتوكولين السابقين يفترض أن توقيتيهما متزامن (نفس التوقيت)، الهجوم الذي يمكن أن يحدث على كليهما هو أن الدخيل يمكن أن يقوم بمحاولة التلاعب بتوقيت B لتقمص شخصية A لـ B بمعاودة إرسال رسالة أرسلت سابقًا بينهما.

في كل البرتوكولات السابقة استعملنا مفاتيح التشفير التماثلي، وبالطبع لو استطاع الدخيل الحصول على المفتاح من خلال وصوله لقاعدة بيانات المفاتيح، فإنه بإمكانه تقمص شخصية أي أحد من أصحاب هذه المفاتيح. في البرتوكولين التاليين سنعتمد على مفاتيح التشفير العام.

- البرتوكول السادس:

$$\boldsymbol{A \rightarrow B: A}$$
$$\boldsymbol{B \rightarrow A: N_B}$$
$$\boldsymbol{A \rightarrow B: \{N_B\}_{K_A^{-1}}}$$

يستطيع الدخيل ويمكن أن يكون B نفسه أن يولد بيانات كما يريد، ثم يرسلها على أنها رقم التحدي فيقوم A بالتوقيع إلكترونيّاً على هذه البيانات بدون علمه، إذ يحسب أن الرقم عشوائي، ويقصد به التأكد من الهوية ثم يستعمل B هذا التوقيع لصالحه.

- البرتوكول السابع:

$$\boldsymbol{A \rightarrow B: A}$$
$$\boldsymbol{B \rightarrow A: \{N_B\}_{K_A}}$$
$$\boldsymbol{A \rightarrow B: N_B}$$

في هذا البرتوكول يستطيع الدخيل ويمكن أن يكون B نفسه أن يطلع على رسالة أرسلت إلى A من طرف آخر من خلال إرساله الرسالة المشفرة نفسها كرقم تحدٍ مشفر فيقوم A بفك التشفير، ويرسل الرسالة إلى B ، وهكذا تمكن B من انتهاك خاصية السرية من خلال هذا الهجوم.

يكمن الحل لهذين البرتوكولين الأخيرين في تحديد هيكلة رقم التحدي الذي يقع إمضاؤه، أو فك تشفيره بين طرفي الاتصال.

نستنتج من هذا الهجومات على أن البرتوكولات البسيطة يمكن أن تؤدي إلى مسائلَ معقدة وخطيرة، وأن التحليل السطحي للبرتوكول يمكن أن لا يتفطن للهجومات المحتملة، ولهذا لا بد من التحليل المنطقي للبرتوكول، وهذا هو توجه كثير من مراكز البحث في العالم المهتمة بأمن برتوكولات الأمن.

نعرض فيما يلي من الفقرات إلى التعرف على برتوكولي أمن مشهورين، وهما: برتوكول Kerberos وبرتوكول أمن الانترنت IPSec.

6.4 بروتوكول التأكد من الهوية Kerberos

يتيح هذا النظام للأطراف المتصلة على شبكة غير آمنة إثبات هوياتهم بعضهم لبعض بطريقة آمنة. طور البرتوكول معهد MIT بأمريكا ونشر النسخة الرابعة منه في عام (1989)، ومازالت قيد الاستعمال في كثير من البرامج ثم النسخة الخامسة في عام (1993) وهي الآن النسخة المعيارية. يعتمد البرتوكول على عمارة العميل-الخادم ويقوم بإثبات متبادل للهوية بينهما، ويعتمد على التشفير التماثلي، ويتطلب تنفيذه وجود جهة موثوقة لتوزيع المفاتيح تسمى Key Distribution Center (KDC). تتكون الأخيرة من خادمين أساسيين، وهما: خادم التأكد من الهوية Authentication Server AS))، وخادم إصدار البطاقات(Ticket Garanting Server TGS). فبرتوكول Kerberos يعتمد على هذه البطاقات في إثبات هوية المستخدمين. تحتفظ KDC بنسخة من المفتاح السري لكل مستخدم. ويحتفظ المستخدم فقط بمفتاح واحد، وهو هذا المفتاح المشترك مع KDC. لكل اتصال بين مستخدمين تقوم KDC بتوليد مفتاح جلسة خاص بهما لاستعماله في تشفير الرسائل المتبادلة بينهما في هذه الجلسة، يشترط برتوكول Kerberos توافق وتزامن الوقت عند أطراف الاتصال ويعتمد على بطاقات تأكيد الهوية التي يصدرها في الإثبات المتبادل لهوية طرفي الاتصال، وأهم خدمة يقدمها Kerberos أنه يجعل الشبكة للمستخدم بمثابة حاسوب واحد، فعندما يتأكد من هوية المستخدم عند أول دخوله على حسابه فإنه يستطيع أن يتصل بشكل آمن بكل الخدمات الموجودة على الشبكة المحلية كخدمة telenet وftpg و rlogin وrsh وغيرها، من دون أن يعيد المستخدم في كل مرة كلمة السر للدخول على هذه الخدمات بل أن برتوكول Kerberos يريحه من ذلك بشكل تام ومخفيٍّ، إذ يجعل حاسوب المستخدم هو الذي يقوم بذلك آليا عوضًا عن المستخدم الحقيقي.

يسعى Kerberos لتحقيق المتطلبات التالية:

1. **الأمن:** لا يمكن للدخيل أن يتقمص شخصية طرف من أطراف الاتصال
2. **الاعتمادية:** باعتبار أن النظام سيستخدم في خدمة مراقبة الدخول للخدمات فلا بد أن يكون قليل العطب وأن يكون هناك طرق استرجاع سريعة وإلا تعطلت كل الخدمات الشبكية.
3. **الشفافية:** لا يدخل المستخدم كلمة سره إلا مرة واحدة، وأما دخوله على بقية الخدمات الشبكية فلا بد أن يكون التأكد من هويته بشكل آلي من خلال Kerberos وبدون أن يشعر المستخدم بذلك.
4. **القدرة الاتساعية:** لا بد أن يكون النظام قادرًا على خدمة أكبر عدد من المستخدمين، وتمكينهم من استعمال أكبر عدد من الخوادم والخدمات الشبكية بدون أن يؤثر ذلك كثيرًا في فاعلية أدائه.

كما يعتمد Kerberos على برتوكول Needham-Schroeder التالي، إلا أنه عوض إرسال أرقام التحدي يرسل الطابع الزمني للتأكد من حداثة مفتاح الجلسة المولد وصد هجوم التكرار الذي يستعمل مفتاح جلسة سابقة.

$$\mathbf{1.}\ A \rightarrow S: A, B, N_1$$

$$\mathbf{2.}\ S \rightarrow A: \{N_1, B, K_{AB}, \{K_{AB}, A\}_{K_{BS}}\}_{K_{AS}}$$

$$\mathbf{3.}\ A \rightarrow B: \{K_{AB}, A\}_{K_{BS}}$$

$$\mathbf{4.}\ B \rightarrow A: \{N_2\}_{K_{AB}}$$

$$\mathbf{5.}\ A \rightarrow B: \{N_2 - 1\}_{K_{AB}}$$

1. 4. 6 عمارة النسخة الرابعة من Kerberos

تتكون عمارة النسخة الرابعة من Kerberos من:

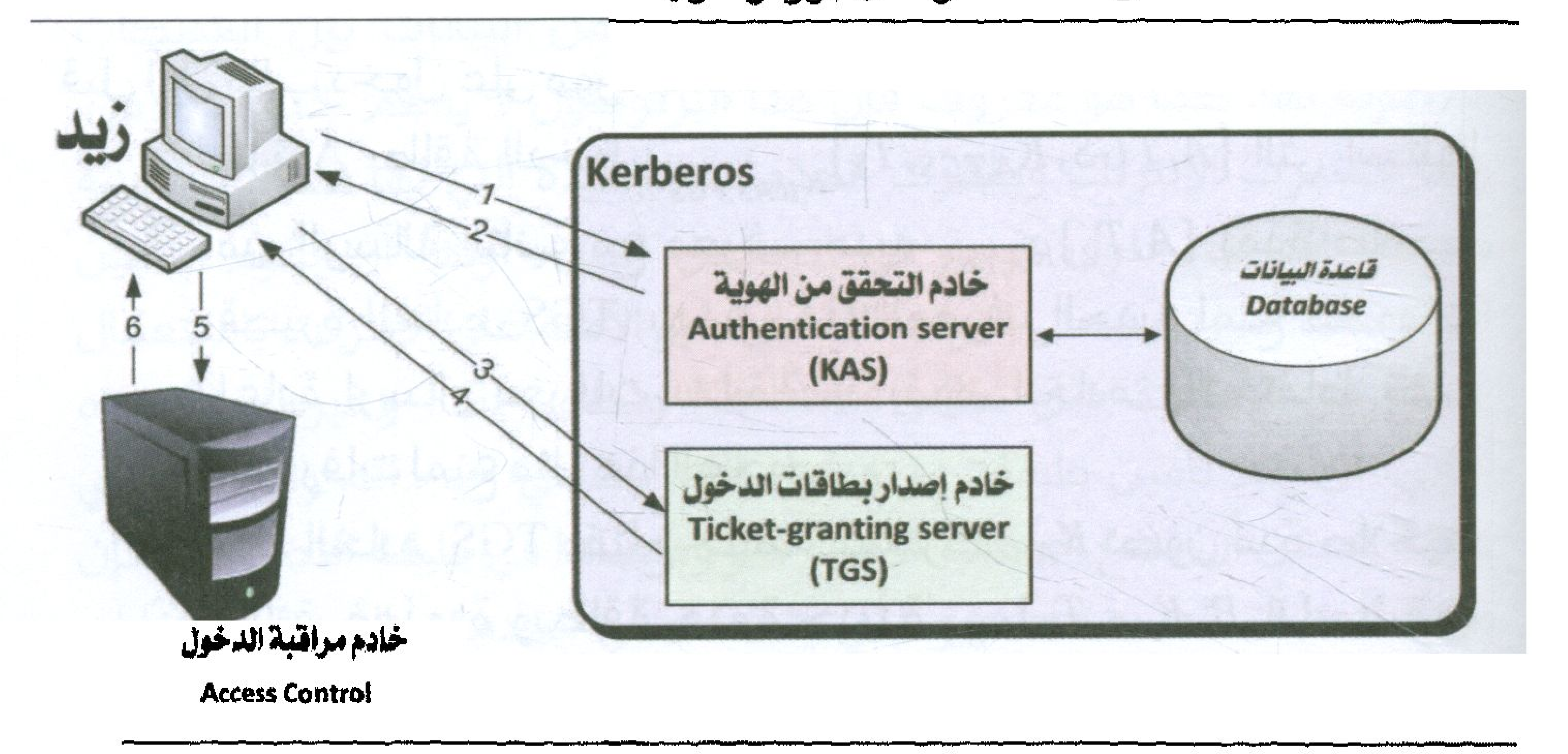
1. خادم Kerberos للتحقق من الهوية (Kerberos Authentication Server KAS) الذي يقوم بخدمة التحقق من الهوية.
2. خادم إصدار بطاقات الدخول (Ticket Garanting Server TGS) الذي يقوم بخدمة إصدار بطاقات السماح بالدخول للموارد.

3. خادم مراقبة الدخول Access Control الذي يقوم بالتحقق من صلاحية بطاقات الدخول للموارد.

2. 4. 6 بروتوكول النسخة الرابعة من Kerberos

يتكون البرتوكول من ست رسائل لثلاث مراحل متتالية، وهي مرحلة التحقق من الهوية، (الرسالة 1 و2) ومرحلة إصدار بطاقة السماح بالدخول، (الرسالة 3 و4) ومرحلة الخدمة والسماح بالدخول لها (الرسالة 5 و6)، (انظر الشكل 25.3).

شكل 3. 25 مراحل بروتوكول Kerberos v4



1. 2. 4. 6 مرحلة التحقق من الهوية

$$1. A \rightarrow KAS: A, TGS$$

$$2. KAS \rightarrow A: \left\{K_{A,TGS}, TGS, T_1, \{A, TGS, K_{A,TGS}, T_1\}_{K_{KAS,TGS}}\right\}_{K_{AS}}$$

- يقوم A بالدخول إلى حاسوبه ويطلب الدخول موارد من الشبكة.
- يقوم الخادم KAS بالدخول لقاعدة البيانات ويرسل إلى A مفتاح جلسة $K_{A,TGS}$ وبطاقة دخول مشفرة $\{A, TGS, K_{A,TGS}, T_1\}_{K_{KAS,TGS}}$

- يبقى المفتاح $K_{A,TGS}$ صالحًا لساعات معينة، حسب نوعية الخدمة التي طلبها A ويكون مولدًا من كلمة سره. كما أن كلا مفتاحي المستخدم والخادم تكون مخزنة في قاعدة بيانات.
- يكتب A كلمة سره في الحاسوب لفك شفرة الرسالة التي تخزن لإتمام الجلسة التي تنتهي بانتهاء صلاحية مفتاح الجلسة $K_{A,TGS}$.

6 .4 .2 .2 مرحلة إصدار بطاقة السماح بالدخول

$$3. A \rightarrow TGS: \{A, TGS, K_{A,TGS}, T_1\}_{K_{KAS,TGS}}, \{A, T_2\}_{K_{A,TGS}}, B$$

$$4. TGS \rightarrow A: \{K_{AB}, B, T_3, \{A, B, K_{AB}, T_3\}_{K_{BS}}\}_{K_{A,TGS}}$$

قبل أول طلب دخول على مورد الشبكة B

- يبرز A بطاقة الدخول $\{A, TGS, K_{A,TGS}, T_1\}_{K_{KAS,TGS}}$ التي استلها من الرسالة الثانية مع معرف جديد $\{A, T_2\}_{K_{A,TGS}}$ بمدة صلاحية قصيرة للخادم TGS. يهدف هذا المعرف الجديد لمنع هجومات إعادة إرسال معرفات سابقة، إذ يقوم الخادم بالاحتفاظ بهذه المعرفات لمنع مثل هذا الهجوم.
- يولد الخادم TGS مفتاح جلسة مشترك K_{AB} تكون مدة صلاحيته دقائق معدودة وبطاقة خدمة جديدة $\{A, B, K_{AB}, T_3\}_{K_{BS}}$ مشفرة بمفتاح مشترك بين مورد الشبكة و خادم TGS

6 .4 .2 .3 مرحلة السماح بالدخول لمورد الشبكة أو الخدمة

$$5. A \rightarrow B: \{A, B, K_{AB}, T_3\}_{K_{BS}}, \{A, T_4\}_{K_{AB}}$$

$$6. B \rightarrow: \{T_4 + 1\}_{K_{AB}}$$

للدخول على مورد الشبكة أو الخدمة B

- يستخلص A المفتاح K_{AB} من الرسالة الرابعة ويرسل لـB بطاقة الخدمة $\{A, B, K_{AB}, T_3\}_{K_{BS}}$ ومعرف جديد مشفر بالمفتاح $\{A, T_4\}_{K_{AB}}$
- يقوم B بفتح بطاقة الخدمة ويستخرج المفتاح المشترك ويرسل لِ A$\{T_4 + 1\}_{K_{AB}}$ كدليل على هويته وأنه B لأنه استطاع فتح البطاقة المرسلة إليه.

ومن خلال هذه الثلاث مراحل يتم التأكد من هوية طرفي الاتصال، والسماح للمستخدم من استعمال موارد الشبكة بشكل مرن وآمن. ومع هذا فإن النسخة الرابعة لبرتوكول Kerberos تعاني من بعض القصور، ككون الرسالة رقم واحد يمكن أن تستخدم كوسيلة لهجوم إنكار الخدمة على الخادم *KAS* وأيضا هنا تشفير مكرر في الرسالة الثانية، وقع تدارك هذه الأشياء وغيرها في النسخة الخامسة من البرتوكول التي تم اعتمادها كنسخة معيارية للبرتوكول.

6.5 برتوكول أمن الانترنت IPSec

تعتمد الانترنت على برتوكول TCP/IP لنقل البيانات بين الشبكات المكونة لها. كما هو معروف فإن هذا البرتوكول لا يدعم خدمات الأمن، ولما انتشرت الإنترنت وظهرت الخدمات الجديدة التي لها متطلبات أمنية معينة، ظهرت حلول تقنية في اتجاهين اثنين: أولها أن نقوم بتأمين التطبيقات التي تنفذ على قنوات الاتصال غير الآمنة على الإنترنت باستعمال جملة من الأنظمة مثل نظام Kerberos ونظام PGP وغيرها. والاتجاه الثاني: أن يتم تأمين طبقات برتوكول الإنترنت في حد ذاته، ولكن في أي طبقة يمكن أن نضع خدمات الأمن؟ فلو وضعناها في طبقة التطبيقات فإن مزية ذلك أن المستخدم هو الذي يتحكم في الخدمات الأمنية، وأيضًا لن نغير شيئًا في الطبقات الأخرى، ولكن المستخدم أيضًا يضطر في كل تطبيق من دمج الخدمات الأمنية بنفسه مما سيعقد عملية تطوير التطبيقات على الانترنت. ولو وضعنا الخدمات الأمنية ما بين طبقة التطبيقات وطبقة النقل مثل حل SSL فإن الميزة في ذلك أن نظام التشغيل لا يقع فيه تغيير، وسيقع تغيير بسيط في التطبيقات، ولكن المشكلة في التعامل مع برتوكول TCP فمثلا SSL يمكن أن لا تقبل بيانات يقبلها برتوكول TCP فيقع إلغاء اتصالات TCP مما يمكن أن يؤدي إلى هجوم إنكار الخدمة. لو وضعنا الخدمات الأمنية في طبقة التوجيه فإن طبقة النقل لن تتغير، وهذا جيد، ولكن التأكد من الهوية سيكون من عناوين الانترنت فقط، لا المستخدمين أنفسهم؛ لأن هذه الطبقة لا اطلاع لديها على ما تحويه الطرود

من بيانات عن التطبيقات. فيما يلي نتعرض لهذا الحل الأخير من خلا برتوكول أمن الانترنت IPSec.

يوفر برتوكول IPSec قنوات آمنة لكل التطبيقات التي تضمن خدمتي التشفير والتأكد من هوية البيانات، كما أنه يمكن أن يقوم بتصفية الطرود اعتمادًا على سياسة أمنية معينة فيعمل عمل جدران الحماية. يقع تنفيذ البرتوكول في نظم التشغيل وفي بوابات الأمن مثل: جدران الحماية والموجهات كما يستعمل في الشبكات الافتراضية الخاصة. يتكون IPSec من ثلاث مكونات هي:

1. ترويسة التأكد من الصحة Authentication Header AH الذي يقوم بتحقيق سلامة وصحة طرود الانترنت ولكن لا يوفر السرية لها.
2. التغليف الآمن للبيانات Encapsulating Security Payload ESP الذي يوفر السرية، وأيضًا السلامة كخيار يمكن استعماله أيضًا.
3. إدارة المفاتيح باستعمال برتوكول تبادل المفاتيح Internet Key Exchange IKE

يقوم IPSec بربط علاقة آمنة بين طرفي الاتصال باستعمال رابطة الأمن Security Association التي من خلالها يتم الاتفاق على الخوارزميات التي سيتم استعمالها بين المرسل والمستقبل، ومفاتيح التشفير وغيرها، وترسل هذه المعلومات من خلال حقول AH/ESP ، ويتم تبادلها عن طريق برتوكول IKE، وتخزينها في قاعدة بيانات رابطات الأمن.

1. 5. 6 ترويسة التأكد من الصحة Authentication Header AH

يتم إضافة ترويسة التأكد من الصحة بين برتوكول TCP وبرتوكول IP لتوفير المعلومات عن رابطة الأمن، كما توفر الترويسة سلامة البيانات وتحمي ترويسة برتوكول IP. يبين الشكل الموالي هيكلة الترويسة وأهمها حقل SPI الذي يحمل بيانات تعريف رابطة الأمن، وحقل رقم تسلسلي لمنع هجوم إعادة إرسال رسالة سابقة، وحقل التأكد من الصحة والسلامة الذي يحمل البصمة MAC المولدة بخوارزمية من خوارزميات توليد البصمات كـ MD5 أو SHA-1.

شكل 26.3 هيكلة طرد AH

← 32 bits →

Next header type of payload after AH	**Payload length** length of AH in 32-bit words (-'2')	**Reserved** for future use
Security Parameters Index (SPI) field identifying the SA for the datagram (value 0 indicates that no SA exists)		
Sequence number field counter value, used to detect replayed packets		
Authentication data variable number of 32-bit words containing the authentication data, e.g., a MAC (MD5 or SHA-1)		

يمكن تنفيذ AH بوضعين اثنين، وهما: وضع النقل (Transport mode)، ووضع النفق (Tunnel mode) (انظر الشكل الموالي).ففي وضع النقل يقع إدراج AH بعد ترويسة IP وقبل بياناته ويقع حساب البصمة على كل الطرد. وفي وضع النفق يقع إدراج AH قبل ترويسة IP ثم نقوم بتغليف الكل بترويسة IP جديدة يمكن أن تحتوي على عناوين IP جديدة مثل: عنوان جدار الحماية أو بوابة الأمن. كما يقع حساب البصمة على كل الطرد بما فيه ترويسة التغليف.

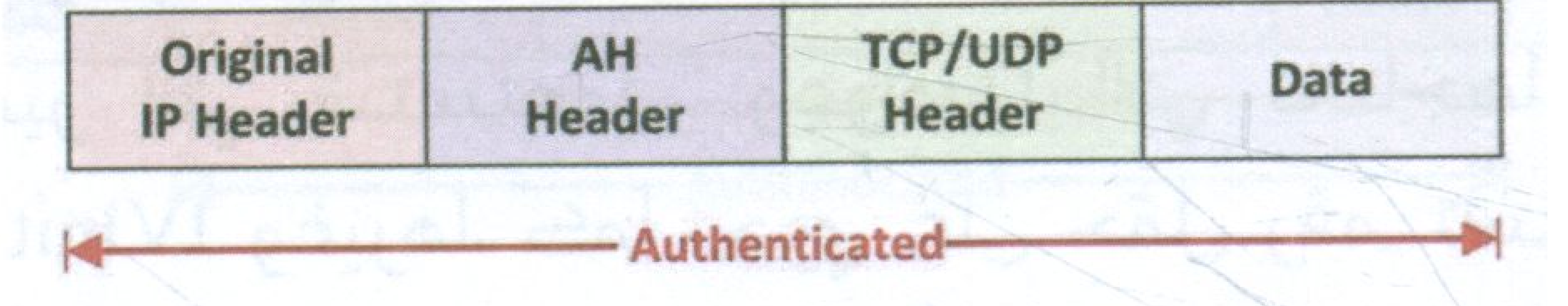
شكل 27.3 أوضاع AH

IPSec Authentication Header (AH): IP protocol number 51

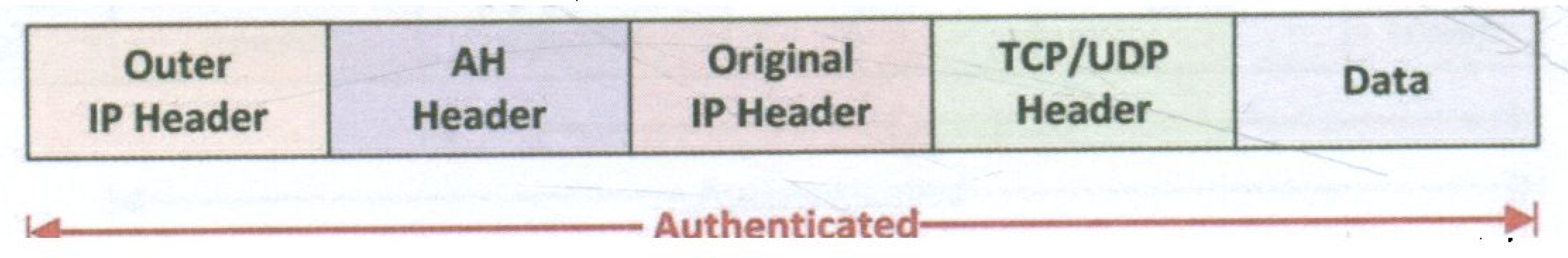
Before applying AH



IPSec Transport Mode: After applyinh AH

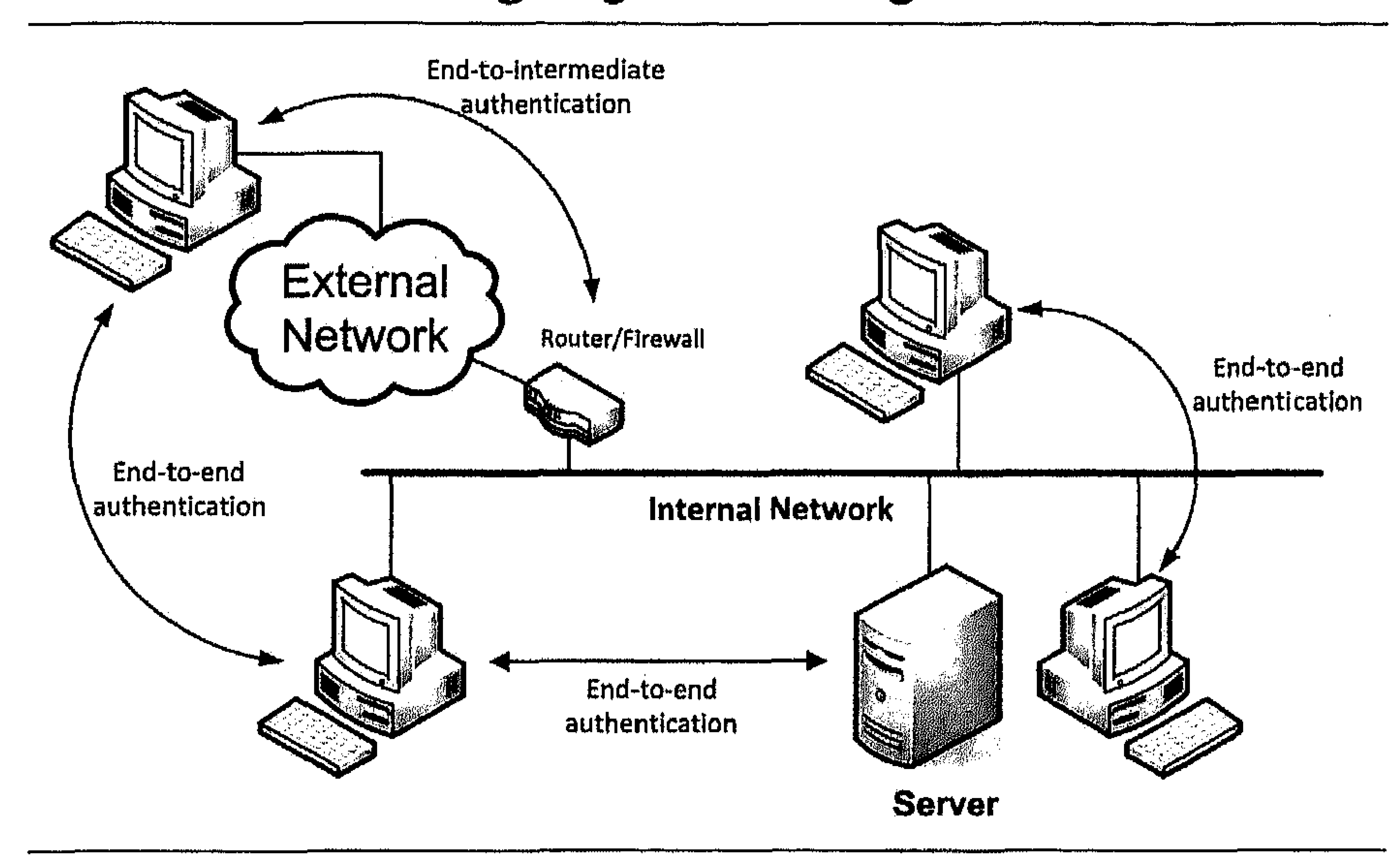


IPSec Tunnel Mode: After applyinh AH



نستعمل عادة وضع النقل في AH في التأكد صحة القنوات بين طرفي اتصال end-to-end)) أما بين طرف اتصال مع بوابة أمن أو جدار حماية فنستعمل عادة وضع النفق (انظر الشكل 27.3)

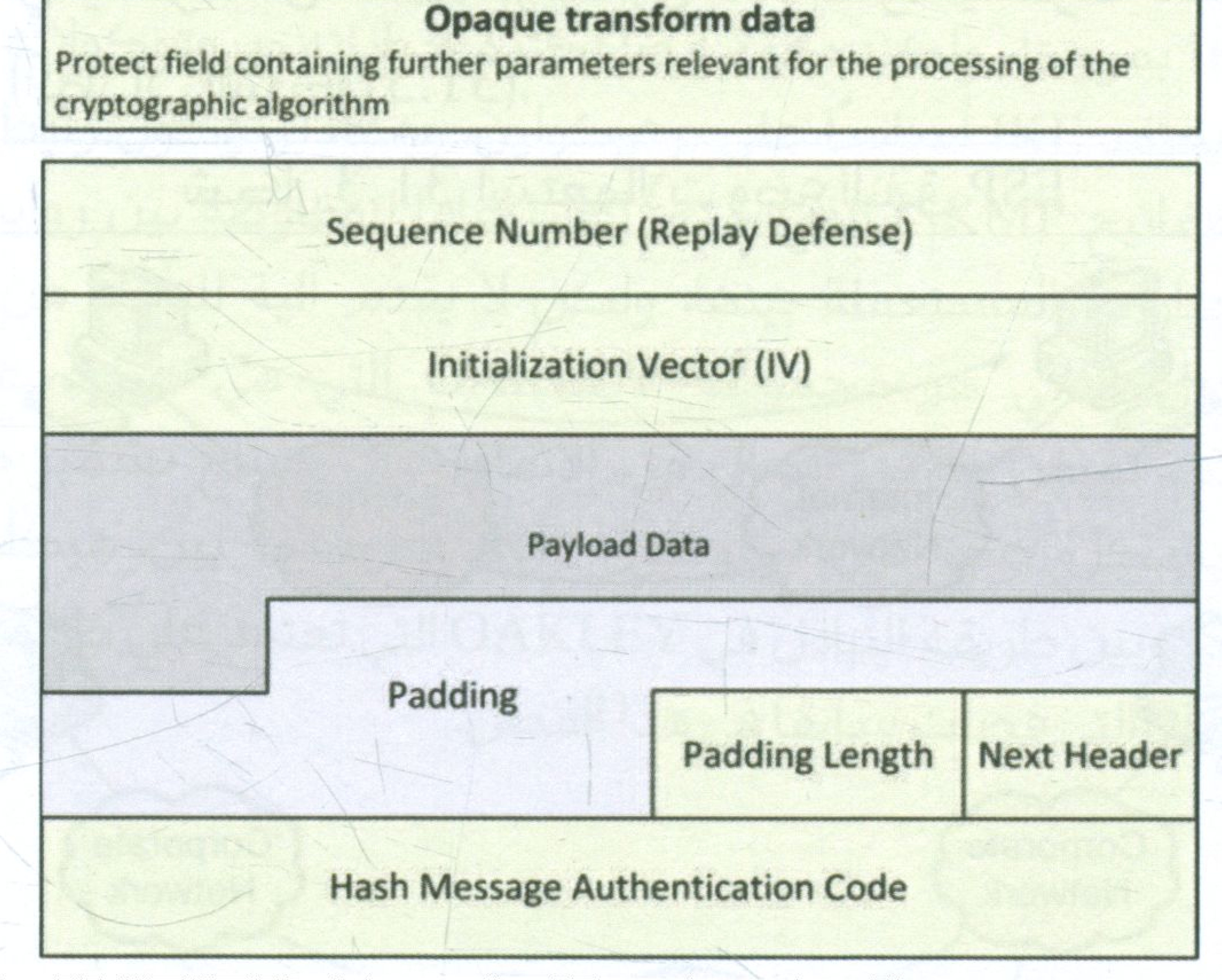
شكل 3. 28 استعمال النفق AH



2. 6.5 التغليف الآمن للبيانات Encapsulating Security Payload ESP

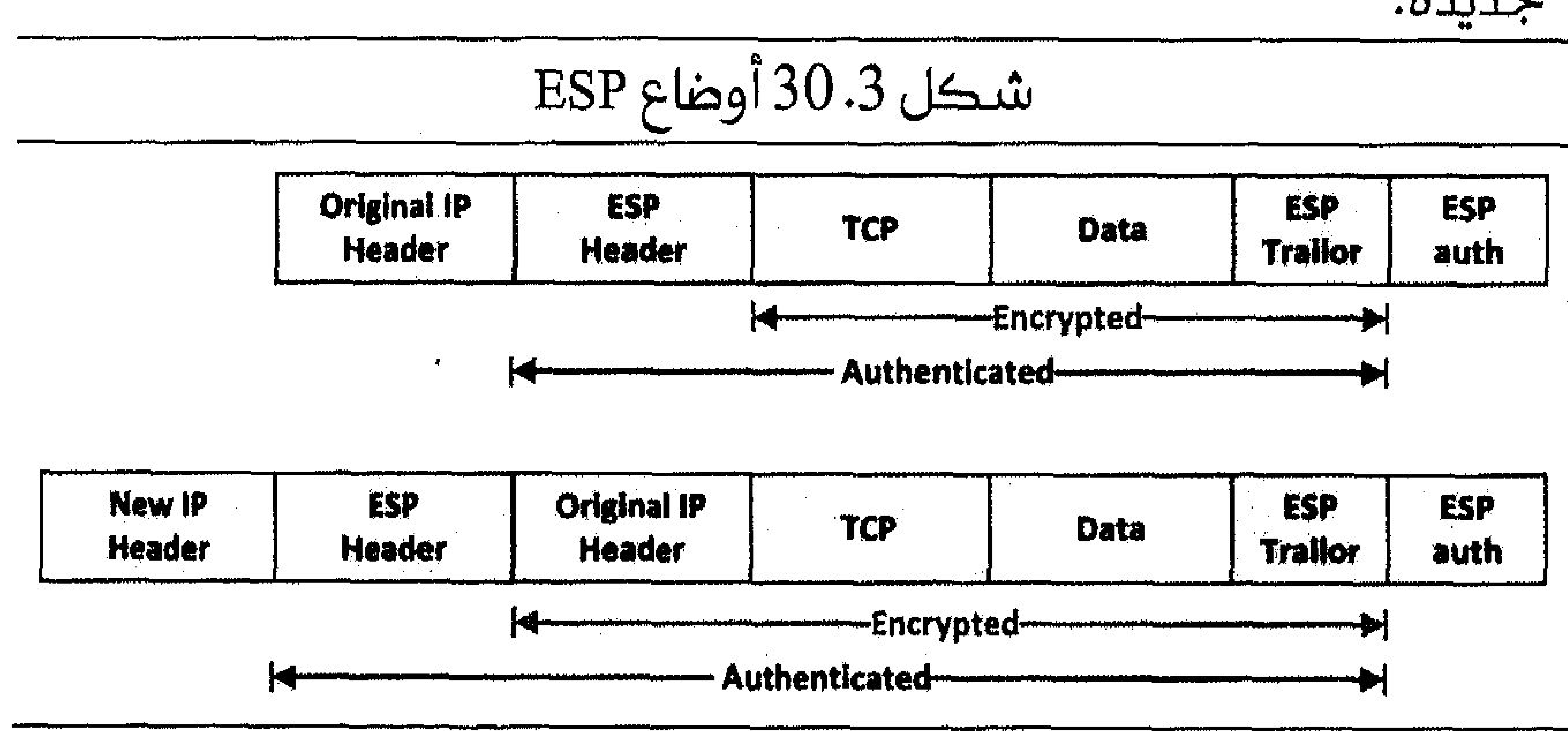
تقوم ترويسة ESP بتوصيف طريقة التشفير وطريقة التأكد من السلامة. يبين الشكل الموالي هيكلة الترويسة، وأهمها حقل SPI الذي يحمل بيانات تعريف رابطة الأمن، وحقل محمي يحتوي على عوامل أخرى تحدد خوارزميات التشفير التي ستستعمل، وعواملها التي تحتاجها مثل IVInitialization Vector وغيرها، كما تحوي على حقل رقم تسلسلي لمنع هجوم إعادة إرسال رسالة سابقة، وكود التأكد من السلامة HMAC.

شكل 29.3 هيكلة طرد ESP



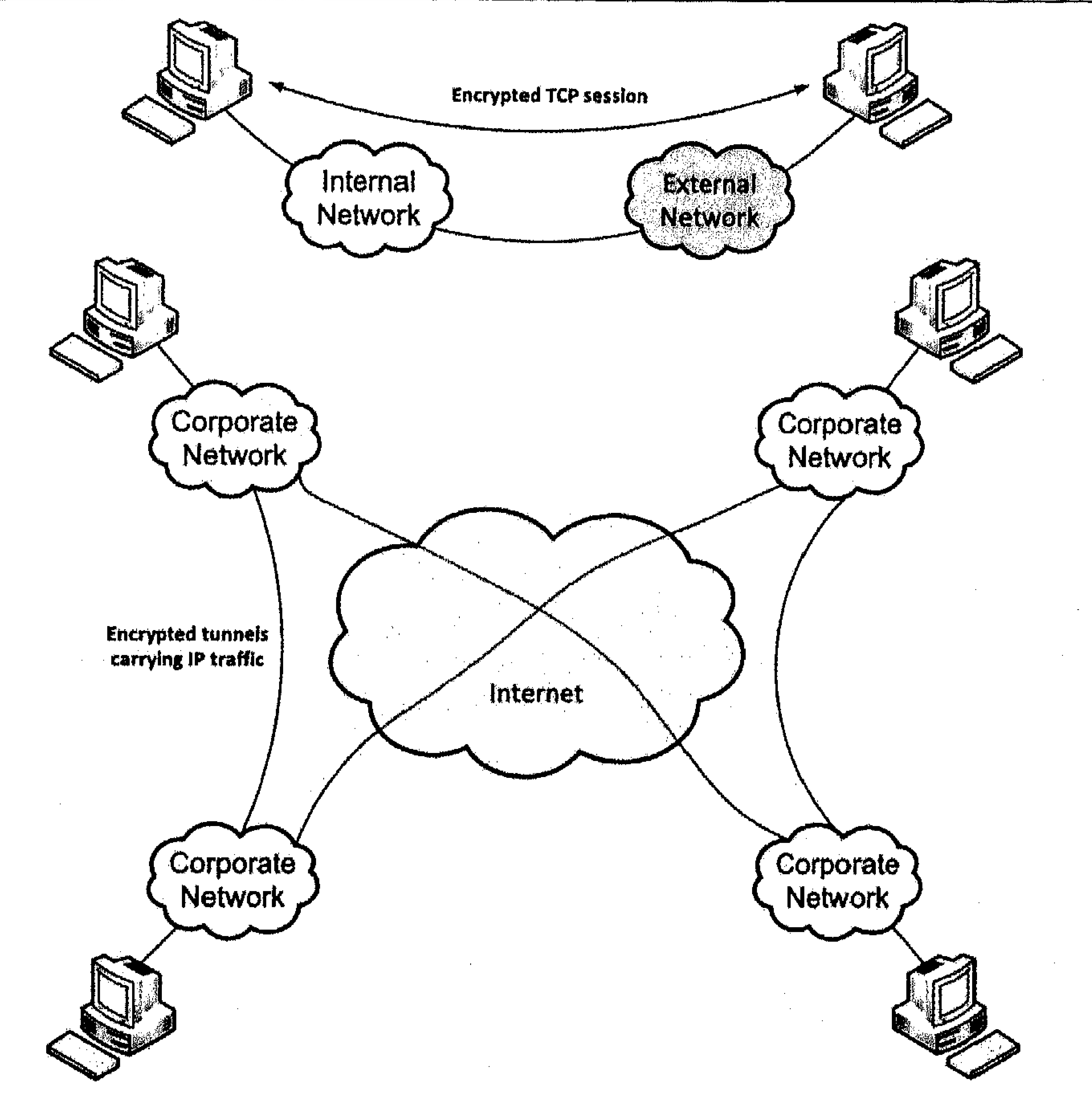
يمكن تنفيذ ESP بوضعين اثنين، وهما وضع النقل(Transport mode) ووضع النفق (Tunnel mode) (انظر الشكل 30.3).ففي وضع النقل يقع تشفير البيانات فقط دون عناوين IP. وفي وضع النفق يقع تغليف الكل بترويسة ESP وهنا يقع تشفير الكل – أي البيانات والعناوين –، ويقع حساب البصمة على الناتج ككل ثم نقوم بتغليف الكل بترويسة IP جديدة.

شكل 30.3 أوضاع ESP



نقوم باستعمال وضع النقل لتأمين اتصال سري بين طرفي اتصال (end-to-end) تدعم بروتوكول IPSec. أما وضع النفق فيستعمل في إنشاء شبكات افتراضية شخصية Virtual Private Network VPN بين بوابات الأمن التي يشترط أن تدعم بروتوكول IPSec ولا يشترط ذلك لطرفي الاتصال (انظر إلى الشكل 31.3).

شكل 31.3 استعمالات وضع النفق ESP



3. 5. 6 بروتوكول تبادل المفاتيح Internet Key Exchange IKE

يقوم هذا البروتوكول بإنشاء روابط الأمن بين أطراف الاتصال على الشبكة، حيث يتم تحديد نوع البروتوكول المستعمل AH/ESP وخوارزميات التشفير وتوليد البصمة، والمفاتيح وغيرها من عوامل الأمن. يتميز IKE بمرونته، ولكنه معقد أيضًا لكثرة خياراته وبروتوكولاته الجزئية وغيرها. يعتمد IKE أساسًا على بروتوكولين، وهما: بروتوكول رابطة الأمن، وإدارة المفاتيح ISKMP الذي يقدم إطارًا عامًا للمفاوضة بين روابط الأمن حول الخيارات المستعملة بينها، ولكن لا يقدم آلية للتأكد من الهوية، والبروتوكول الثاني هو مجموعة OAKLEY التي هي عبارة عن عدة بروتوكولات تمكن طرفي اتصال من الاتفاق على مفتاح تشفير مشترك بينها. وبشكل عام فإن بروتوكول IKE يمزج ما بين هيكلة طرود ISAKMP وبين طريقة التبادل في OAKLEY التي تعتمد على طريقة ديفي–هيلمن DHالتي فصلت سابقًا في هذا الفصل.

7- مراجع إضافية

7.1 كتب

ننصح بالرجوع إلى الكتب التالية:

- Bruce Schneier. Applied Cryptography. John Wiley & Sons, New York, 1996.

- Dieter Gollmann. Computer Security. Wiley, 2000.

- Alfred J. Menezes, Paul C. van Oorschot, Scott A. Vanstone. Handbook of Applied Cryptography. CRC Press, 1996. Available online at http://cacr.math.uwaterloo.ca/hac/

- Matt Bishop. Computer Security (Art and Science). Pearson, 2003.

- Niels Ferguson and Bruce Schneier. Practical Cryptography. John Wiley &Sons, New York, 2003.

- Williams Stallings. Cryptography and Network Security. Prentice Hall, 2003.

7.2 مواقع

- The International PGP Home Page: http://www.pgpi.org/
- SDSI/SPKI (and PKI and PGP): http://world.std.com/~cme/html/spki.html
- http://en.wikipedia.org/wiki/X.509
- http://en.wikipedia.org/wiki/Message_authentication_code

8 - أهم مصطلحات الفصل

Message Digest	بصمة الرسالة
MAC	كود التأكد من الهوية
MIC	كود التأكد من السلامة
PKI	البنية التحتية للمفتاح العام
CA	جهة مصادقة وإصدار الشهادات
RA	جهة تسجيل الشهادات
Directory	الدليل
Certificate	شهادة
PGP	الخصوصية فائقة الحسن
CRL	قائمة الشهادات الملغاة
Digital Signature	التوقيع الإلكتروني
Key Escrow	نظام حل المفاتيح

9 - تمارين الفصل

1. في نظام دفي-هيلمن (Diffie-Hellman) افترض أن هناك شخصين P_2و P_1 يردان أن يتبادلا مفتاح التشفير السري بينهما حيث اتفقا على اختيار رقمين صحيحين هما $a = 3$ و $b = 5$ ، وبشكل خاص اختار P_1 رقمًا عشوائيًا هو $i = 3$، واختار P_2رقمًا عشوائيًا آخر هو $j = 2$ ،أوجد مفتاح التشفير K_1 لـ P_1 ومفتاح التشفير K_1 لـ P_2،وتحقق من أنهما متطابقان ($K_2 = K_1$)؟

2. متى يمكن اعتبار دالة البعثرة التشفيرية آمنة ؟

3. بين بشي من الإيجاز خصائص الشبكة الشخصية التخيلية (VPN)؟

4. أحد المكونات الأساسية للشبكة الشخصية التخيلية (VPN) هي:

ش. خادم VPN .

ص. خوارزمية التشفير .

ض. نظام تحديد شرعية المتعامل

ط. جميع ما ذكر.

5. ضع دائرة على رقم الإجابة الصحيحة في كل فقرة من الفقرات التالية:

أ. إحدى استخدامات التواقيع الرقمية هو:

1. تبادل المفاتيح.
2. منع الجحود والإنكار.
3. تعمية المعلومات.
4. الحماية ضد الفيروسات.

ب. في نظام تشفير باستخدام المفتاح العام، قام مرسل بتشفير رسالة بواسطة المفتاح العام للمستقبل. ما هو المفتاح الذي يستخدمه المستقبل في فك شفرة الرسالة؟

1. المفتاح الخاص للمرسل.
2. المفتاح العام للمستقبل.
3. المفتاح العام للمرسل.
4. المفتاح الخاص للمستقبل.

ج. في نظام التوقيع الإلكتروني يتم:

1. تشفير الرسالة باستخدام المفتاح الخاص للمرسل، ومن ثم فكها باستخدام المفتاح العام للمرسل.
2. تشفير الرسالة باستخدام المفتاح الخاص للمستقبل ومن ثم فكها باستخدام المفتاح العام للمستقبل
3. تشفير الرسالة باستخدام المفتاح العام للمرسل، ومن ثم فكها باستخدام المفتاح الخاص للمرسل.
4. تشفير الرسالة باستخدام المفتاح الخاص للمرسل، ومن ثم فكها باستخدام المفتاح العام للمستقبل.

6. افترض أن هناك دالة البصمة الإلكترونية للرسالة (message digest function). ذات 128 بيت، وافترض أيضا أن هناك قيمة محددة للبصمة الإلكترونية للرسالة (message digest) هي *d*، وترغب في إيجاد رسالة لها بصمة الإلكترونية للرسالة (message digest) للقيمة *d*. وبمعرفة وجود عدد كبير من الرسائل ذات 2000 بيت التي تقابل ؟؟؟

7. ما أكثر وأقل كمية من الحشو المطلوب لكل دالة من دوال البصمة الإلكترونية للرسالة (message digest functions).

8. وضح أن دالة حساب المجموع (checksum) في خوارزمية MD5 لن تكون دالة بصمة الإلكترونية جيدة للرسالة (message digest function)، وذلك عن طريق بيان كيفية تكوين رسالة باستخدام حساب المجموع (checksum) معطى.

9. بشكل موجز: اشرح فكرة نظام تشفير الجمل (ElGamal) من خلال التالي:

ظ. ما هي الدالة وحيدة الاتجاه في هذا النظام.

ع. ما هو trapdoor في هذا النظام.

غ. حدد المفتاح الخاص والمفتاح العام في هذا النظام.

ف. صف تحقيق الأمان في هذا النظام.

10. في تشفير الجمل (ElGamal)، ماذا يحدث لو أن C1 و C2 تم التبديل بينهما أثناء الإرسال.

11. افترض أن زيدا استخدم المفتاح العام لعبيد في نظام تشفير الجمل (e2 =8) (ElGamal و e1 =2) لإرسال رسالتين P=17 وP'=37 ، وباستخدام نفس الرقم الصحيح العشوائي r=9 ، وقام سعد باقتحام الرسالة المشفرة وبطريقة ما وجد أن قيمة P=17 . بين كيف يمكن لسعد أن يستخدم هجوم الرسالة الأصلية المعروفة لإيجاد قيمة P' .

12. ما الفرق بين سلامة الرسالة وموثوقية الرسالة.

13. بين المعيار الأول والثاني لدالة البعثرة التشفيرية.

14. باستخدام خوارزمية RSA للتشفير، اجعل $p = 809$ ، $q = 751$ و$d = 23$. قم بحساب قيمة المفتاح العام ومن ثم قم بإجراء العمليات التالية:

ق. قم بالتوقيع على الرسالة والتحقق منها باستخدام $M1 = 100$. ارمز لهذا التوقيع بالرمز S_1.

ك. قم بالتوقيع على الرسالة والتحقق منها باستخدام $M2 = 50$. ارمز لهذا التوقيع بالرمز S_2.

ل. أثبت أنه في حال $M = M1 \times M2 = 5000$ ، فإن $S = S_1 \times S_2$

15. باستخدام خوارزمية الجمل (ElGamal) للتشفير، اجعل $d = 700$ ، $p = 881$. أوجد قيم لكل من e_2 و e_1.اختر $r = 17$ ، ومن ثم أوجد قيم كل من S_2 و S_1 عندما $\boldsymbol{M = 400}$.

16. باستخدام خوارزمية تشفير البيانات المعياري DES، اجعل $d = 14$ و $q = 59$ ، $p = 709$ اوجد قيم لكل من e_1 وe_2اختر $r = 13$ ، اوجد قيم كل من S_1 و S_2 عندما $h(M) = 100$ .. ومن ثم تحقق من التوقيع الإلكتروني.

17. قم بإجراء العمليات التالية:

م. في خوارزمية RSA للتشفير، أوجد العلاقة بين حجم s وحجم n .

ن. في خوارزمية الجمل (ElGamal) للتشفير، أوجد حجم كل من S_1 و S_2 نسبة إلى حجم p .

ه. في خوارزمية تشفير البيانات المعياري DES، أوجد حجم كل من S_1 و S_2 نسبة إلى حجم p وq.

الفَصْلُ الرَّابِعُ

أَمْنُ النُّظُمِ وَالتَّطْبِيقَاتِ

يَهْدِفُ هَذَا الفَصْلُ إِلَى:

1. التعرف على نظم جدران الحماية.
2. التعرف على نظم اكتشاف الاختراقات.
3. التعرف على نظم اكتشاف البرامج الخبيثة.

1 - مقدمة الفصل

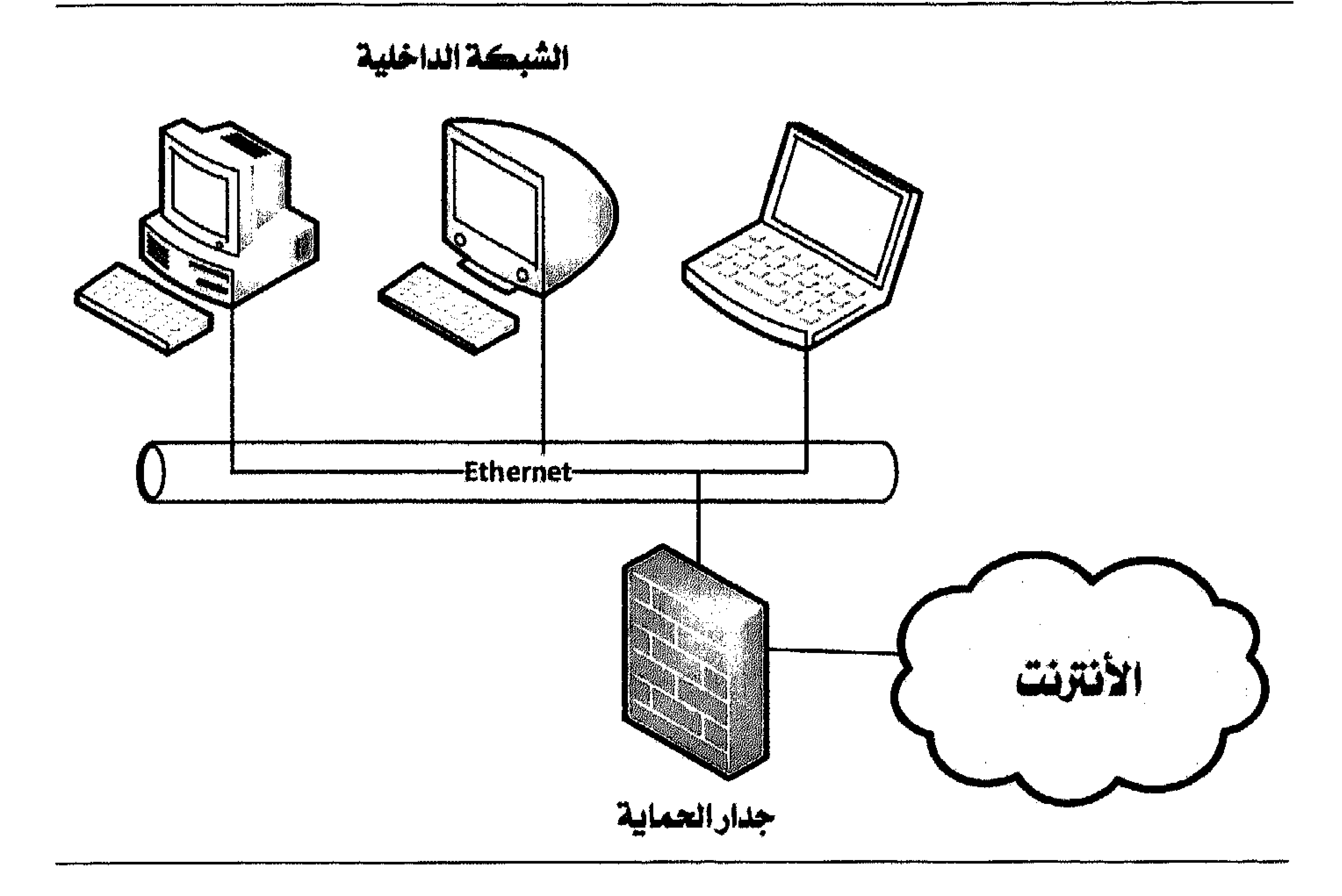
يهدف هذا الفصل إلى التعرف على نظم جدران الحماية، يتعرض الفصل إلى أساسيات تطوير جدران الحماية، وكيفية تطوير سياساتها الأمنية، وكيفية تطوير مجموعة قواعد تصفية الطرود التي هي عمدة أغلب أنظمة جدران الحماية اليوم، كما يناقش كيفية عمل البوابة التطبيقية، أو الخادم الوكيل في الشبكات، وكيفية عملها جنبًا إلى جنب مع جدران الحماية، وأخيرًا كيفية صناعة واستعمال جَرَّاتِ العسل وأهميتها الأمنية.

2 - نظم جدران الحماية

جدار الحماية هو العنصر الأساس للأمن، إذ يعتبر اليوم من أهم أجزاء الحماية الأمنية وبدونه تكون الشبكة عرضة لهجومات خطيرة جدًا خلال ساعة، ولربما دقيقة تنجم عنها أضرار كبيرة. هذا النظام الموجود بالملايين في آلاف الشبكات اليوم هو ذلك الجهاز الذي يحجز بين الشبكة الداخلية، وشبكة الإنترنت بحيث يحد من الهجومات العدوانية من وعلى الشبكة الداخلية. وبشكل عام يلعب جدار الحماية دور الإدارة والتحكم في حزم البيانات بين الشبكات المختلفة.

هناك جملة من الطرق في تطوير جدار الحماية أهمها **مصفي الطرود، والخادم الوكيل أو البوابة التطبيقية**. مصفي الطرود هو نظام بسيط، غالبًا ما يكون في الموجهات ويتعرف على خصائص رؤوس الطرود فقط، إذ يتعرف على مصدر الطرد، ووجهته، وبناءً عليه يتم السماح للطرد بالولوج أو عدمه، وبهذا الاعتبار يكون عمله مُنْضَوٍّ تحت الطبقة الثانية والثالثة في النموذج المعياري للشبكات أوزي لمنظمة الأيزو. وبعض مصفيات الطرود تأخذ بعين الاعتبار معلومات الاتصال والجلسات في حزم البيانات، وبهذا تتعامل أيضًا مع الطبقة الرابعة في الأوزي، وهي طبقة نقل البيانات. وأما الخادم الوكيل أو البوابة التطبيقية فتكون – عَادَةً – في بوابة الشبكة، ويمكنها التعرف على طلبيات العملاء الداخلة والخارجة للشبكة، ومنع الطرود بالنظر إلى محتواها، فهي بهذا تتعامل أيضًا مع الطبقة السابعة من نموذج الأوزي. كما تلعب البوابة التطبيقية دور الوكيل عن العملاء مع الشبكة الخارجية، وبهذا نضمن حجب عناوين الشبكة الداخلية عن الإنترنت والشبكات الخارجية .

شكل 4.1 مكان جدار الحماية



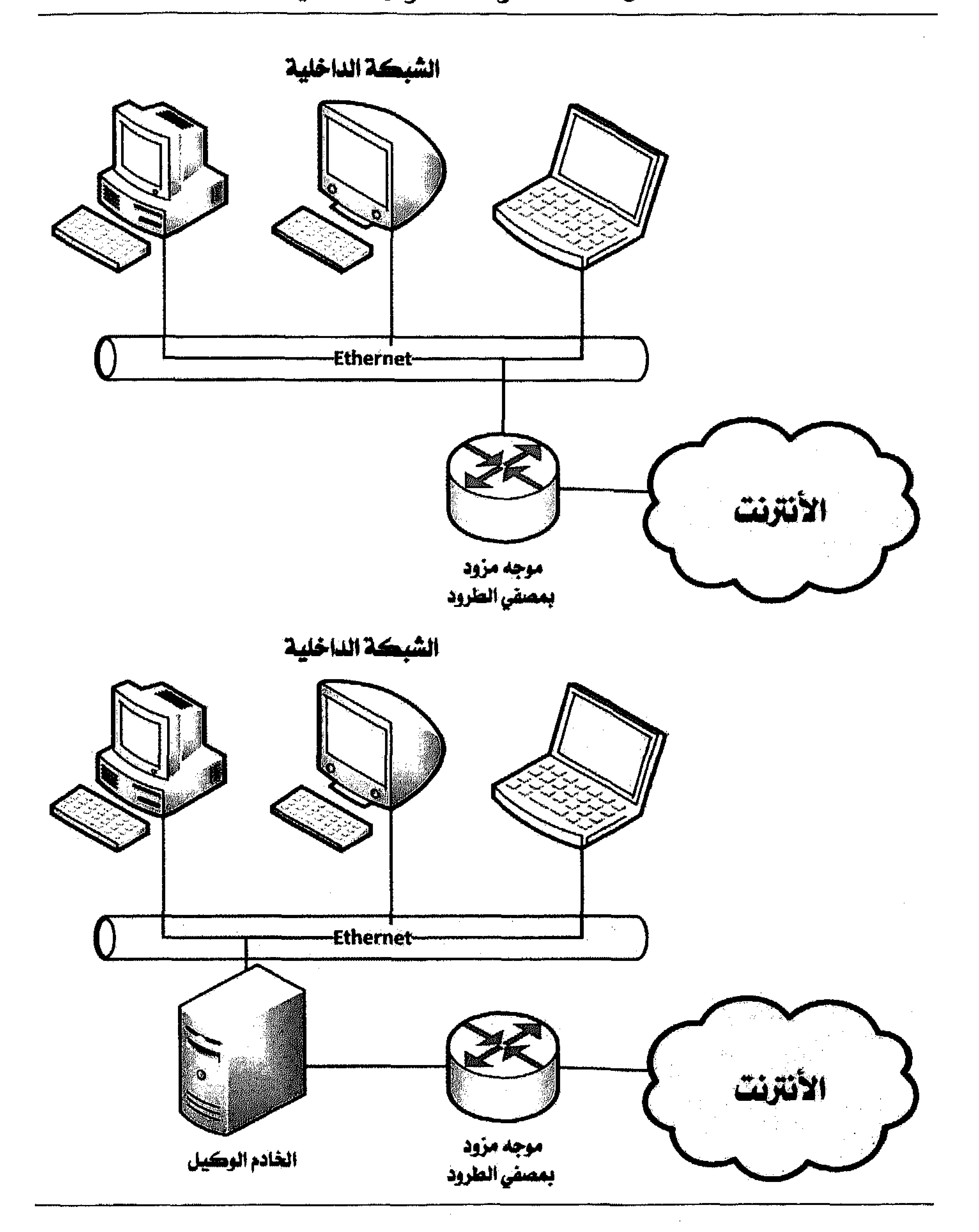
أما جملة ما لا تستطيع جدران الحماية القيام به فيمكن أن نلخصه في خمس نقاط أساسية. أولها التعرف على الفيروسات والأوبئة الرقمية، ومع أن بعض الجدران يقوم بذلك، إلا أن هذا الخيار غير منصوح به ألبتة؛ لأنه يجعل الشبكة بطيئة جدًا، ولهذا يجب استعمال جملة من برامج اكتشاف الفيروسات في الشبكة الداخلية. ثانيها: عدم تحكمه في أخطاء العاملين في المؤسسة، كمن يشغل البرامج المشبوهة، أو يفتح الرسائل البريدية المجهولة مما يجر المشاكل للشبكة، ومن هذا الجنس ما يكون في الحزم التي يسمح بدخولها وَفْقًا لقواعد التصفية، وتحمل في طَيِّهَا هجومًا أو وباءًا رقميًا. ثالثها: الاتصالات الأخرى التي لا تعبر عن طريق جدار الحماية أصلا، كبعض تلك التي تتم عن طريق المودم، أو عن طريق الشبكات اللاسلكية، إذ يمكنها أن تتجاوز الحماية الأمنية، أو جدار الحماية كاملاً ولا تخضع للمراقبة. رابعها: حين يسرب العاملون بصفةٍ مباشرةٍ أو غيرِ

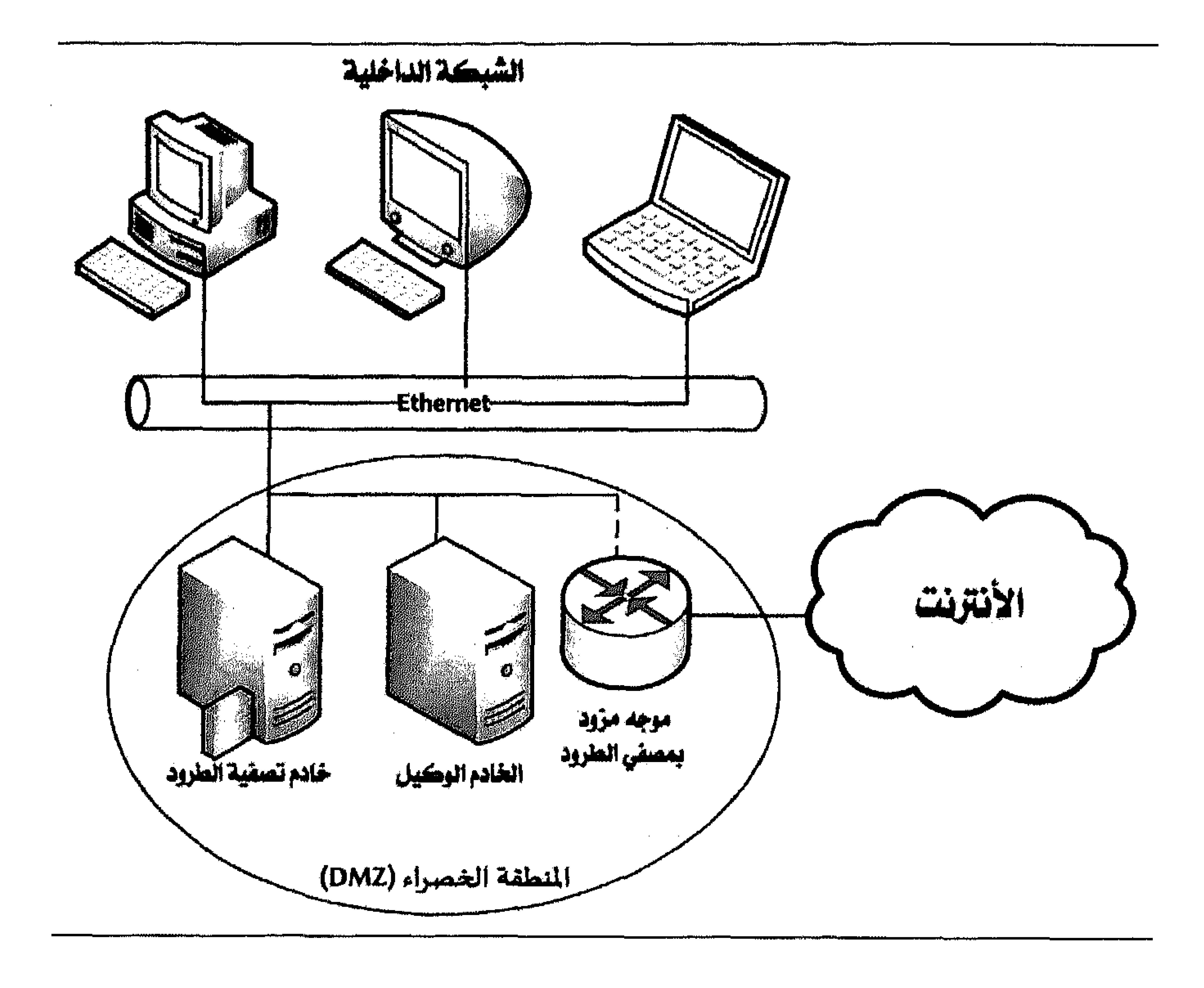
مباشرة قصداً أو بدون قصدٍ جملةً من المعلومات الخطيرة الخاصة للمهاجمين من خلال استعمال الآخرين لتقنيات الهندسة الاجتماعية. وأخيراً لا يستطيع جدار الحماية جبر السياسات الأمنية الضعيفة للمؤسسة ككل، وخصوصاً السياسات الخاصة بجدار الحماية؛ إذ حينها لا يملك الجدار التعليمات الصحيحة في التعامل مع الطرود، ويصبح عملياً غيرَ ذي أهمية.

2.1 خيارات تطوير جدران الحماية

هناك عدة خيارات في تصميم عمارة جدار الحماية في الشبكة. فيمكن أن نستعمل مصفياً واحداً للطرود، حيث يوضع عادة مباشرة بين الشبكة الخارجية والداخلية، ويتم تزويده بقواعد التصفية، وقوائم المراقبة للسماح للطرود، أو الحزم بالولوج أو عدمه. كما يمكن أن نجعل وراء المصفي جهازاً مزوداً بأكثر من بطاقة شبكة يعمل كَوَكِيْلٍ عن الشبكة الداخلية، وبهذا نحصل على نظام أمان أعلى وأكثر اكتمالاً؛ إذ يقوم المصفي بتصفية حزم البيانات الداخلة والخارجة، ويقوم الخادم الوكيل بإخفاء عناوين وطلبيات أجهزة الشبكة الداخلية. أما الخيار الأخير والذي يعتبر أكثر شيوعاً واستعمالاً هو خيار المنطقة المجردة من السلاح أو المنطقة الخضراء، حيث يتكون جدار الحماية أساساً من جهازين اثنين، وهما المصفي المدمج في الموجه المتصل بالانترنت، وخادم تصفية الطرود، ويكون متصلاً بالشبكة الداخلية، حيث يقوم الاثنان بتنفيذ مهام جدار الحماية للشبكة. كما يمكن أن نضع في هذه المنطقة الخضراء خادم وكيل أو خادم ويب، أو بريداً بحيث تكون الخدمة متوفرة للشبكة الخارجية، وتكون بقية الشبكة معزولة، فتكون أكثر أماناً وحمايةً.

شكل 2.4 عمارات جدران الحماية.





2.2 إنشاء سياسة أمن جدران الحماية

هناك خيران اثنان في تحديد أي سياسة أمنية، ومنها: سياسة جدران الحماية، وهما خيارا المنع والمنح، والأخذ بأحدهما يرجع للمؤسسة. فأما خيار المنع فنمنع كل شيء إلا ما احتيج إليه حتمًا. وأما خيار المنح فإننا ننشئ قائمة بالأشياء المسموح للمستخدمين بفعلها، ثم نمنح كل مستخدم منح وصول محددة إليها. أما مصفي الطرود فإنه بعد الفحص إما أن يسمح للطرد بالدخول أو يمنعه من ذلك، مع إرسال إشعار للعنوان المصدر يعلمه بعدم السماح لطرده بالدخول، وإما أن ينبذه ويلغيه دون إشعار المرسل بذلك، وهذا الخيار هو الأسلم لما في الخيار الثاني من كشف معلومات عن مصفي الطرود ونوعه، وعلى أي نظام تشغيل يعمل، بينما الخيار الثالث لا يشعر المهاجم أصلاً بوجود مصفي طرود في الشبكة.

بشكل عام يمكن أن تحوي سياسة أمن جدران الحماية جملة من العناصر أو البنود، أهمها: بند الاستعمال المقبول أو المرضي، وبند الاتصال

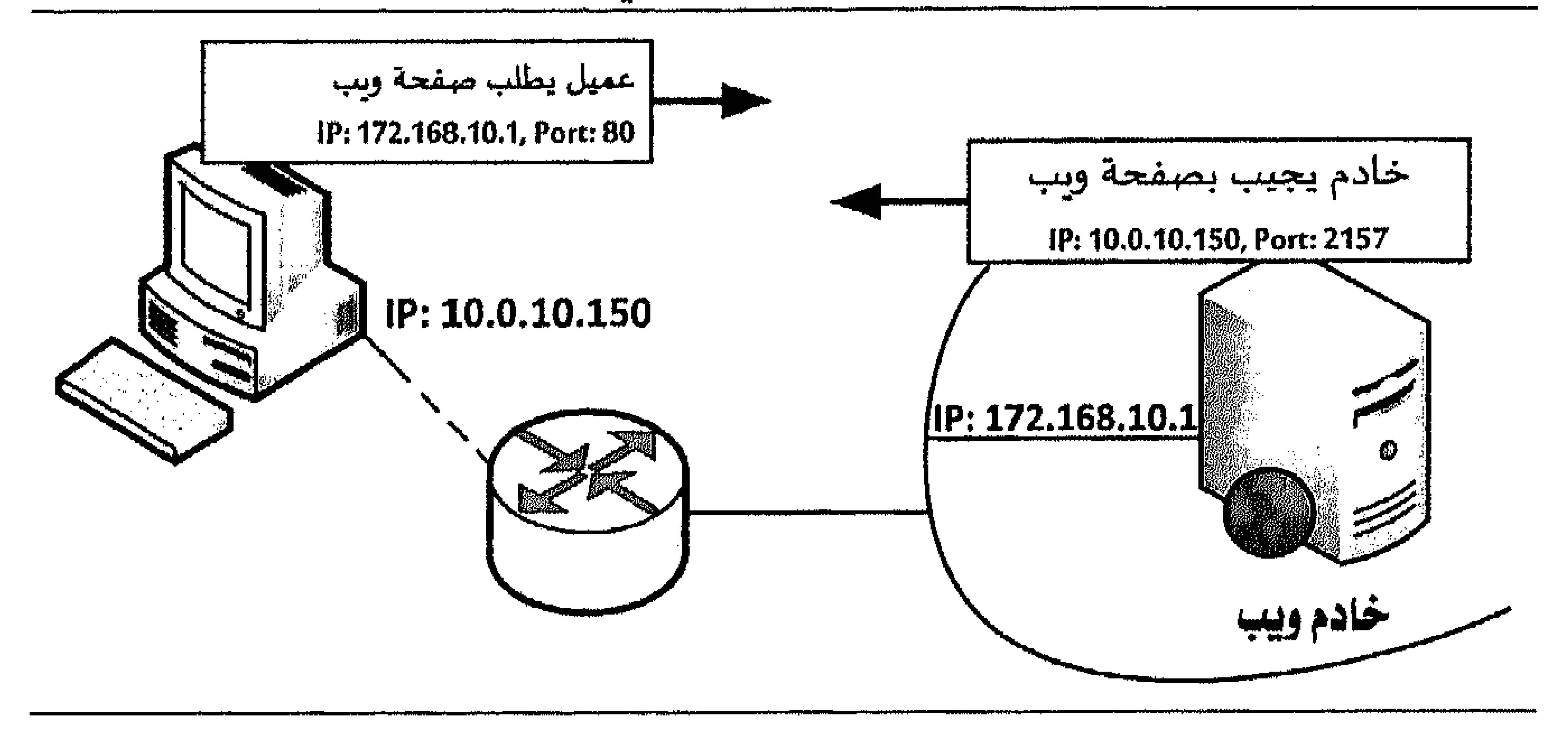
بالشبكة، وبند المستخدمين المتعاقدين، وبند مديري جدران الحماية. في هذه البنود نقوم بتحديد التفاصيل الدقيقة لكل بند فمثلا بالنسبة لبند الاستعمال، نحدد أي البرامج والتطبيقات التي يسمح بتثبيتها، هل يمكن أن تستعمل التطبيقات خارج مجال العمل أو في البيت، وكيفية التعامل مع كلمات السر، وهل البريد يستعمل في المؤسسة فقط أو لا؟ وهكذا. أما بالنسبة لبند الشبكة فنحدد ما إذا يسمح بمسح الشبكة ككل أو لا، وهل يسمح بخدمات البريد والويب؟ وأيضًا من يقوم بتثبيت برامج اكتشاف الفيروسات وتحديثها؟ وغير ذلك. أما بند المتعاقدين فننظر في الموارد التي يمكنون من استخدامها، وهل يمنحون الصلاحيات الكاملة أو صلاحيات محدودة، وهل يمكّنون من مسح الشبكة أو تحميل الملفات أو لا، وفي بند مديري جدران الحماية نحدد أساسًا من يقوم بالإدارة، وهل يتطلب ذلك شهادات معينة، ولمن يقع إرسال تقارير الإدارة وكذلك تحديد المديرين المباشرين على مدى ساعات اليوم والأسبوع.

3. 2 مصفي الطرود وقواعد التصفية

يتموقع مصفي الطرود في الموجه المتصل بالإنترنت، ويعتمد على قواعد تصفية معينة لمراقبة حزم البيانات في الولوج للشبكة الداخلية أو الخروج منها. يمكن أن يعمل مصفي الطرود لوحده أو جنبًا لجنب مع خادم وكيل أو مصفي طرود آخر، كما في عمارة المنطقة الخضراء حيث يكون المهاجم مجبرًا على اختراق الْمُصَفِّيَيْن كِلَيْهِمَا، لتجاوز نظام جدار الحماية. لإنشاء قواعد التصفية لا بد من الإجابة على جملة من المسائل مثل قائمة الخدمات في الانترنت المسموح بالوصول إليها من الشبكة الداخلية، والعكس وما الأجهزة التي تملك صلاحيات خاصة، والأجهزة التي لا تملك ذلك. أما ما يؤخذ بعين الاعتبار في عملية التصفية فيرجع إلى تحديد واجهة التقاط بطاقة الشبكة، ووجهة الطرود، ومجموعة عناوين ومنافذ الإنترنت التي تدخل في توجيه قرارات المصفي، وأيضًا بعض المعلومات على بروتوكلات طبقة التطبيقات.

قبل التمثيل لقواعد التصفية لا بد من التعريف بالمقابس المستخدمة في أي اتصال بين طرفين اثنين على الشبكة.

شكل 3.4 عمارة مصفي الطرود.



فكل جهاز متصل بالإنترنت يملك عنوان انترنت أوحد، يمكن من تحديده في الشبكة، وبما أنه يمكن أن يقدم هذا الجهاز أكثر من خدمة من خدمات الإنترنت، كالويب أو البريد الإلكتروني، فلذلك نحتاج إلى تحديد منفذ الخدمة أيضًا. فبتحديد المقبس – وهو زوج عنوان الإنترنت والمنفذ – نكون قد حددنا الخدمة ككل بشكل دقيق. كل اتصال في الإنترنت يحتاج إلى تحديد طرفي الاتصال من خلال مقبسيهما. فمثلا في خدمة الويب يرسل العميل طلبية إلى خادم الويب بتحديد مقبس الخادم (172.168.10.1،80)، ويجيبه الخادم على مقبس (10.0.10.150،2157) أما المنفذ 80 فهو منفذ عام معلن لكل خادم ويب، وأما المنفذ 2157 فإنه منفذ العميل، ويحصل عليه خادم الويب باستخراجه من طلبية العميل.

يعرض الجدول 1.4 مثالاً من قائمة قواعد التصفية، فالقاعدة الأولى خاصة ببرتوكول TCP للطرود الوافدة إلى الشبكة على المنفذ 80، وهو خادم الويب والقرار هو السماح لها بالدخول، وهذا يعني أننا سنسمح باستقبال طلبيات الشبكة الخارجية على خادم الويب الداخلي، والقاعدة الثانية تمكن خادم الويب بإرسال الأجوبة على هذه الطلبيات، لأن وجهة الطرود إلى الخارج على أي منفذ أكبر 1024. أما القاعدة الثالثة فهي للسماح لمن بالداخل بإرسال طلبيات على خوادم ويب خارجية، والقاعدة الرابعة

لاستقبال أجوبة هذه الخوادم، وأما القاعدة الخامسة فهي منع كل خدمة أخرى عدا ما استثني في الأربع القواعد الآنفة الذكر.

جدول 4. 1 قواعد التصفية في جدار الحماية.

Rule	Direction	Protocol	Src. Address	Dest. Address	Port	Action
1	Inbound	TCP	External	Internal	80	Allow
2	Outbound	TCP	Internal	External	>=1024	Allow
3	Outbound	TCP	Internal	External	80	Allow
4	Inbound	TCP	External	Internal	>=1024	Allow
5	All	All	All	All	All	Disallow

نلاحظ في المثال السابق أن القاعدة الرابعة يمكن أن تسمح لطلبية ما من أي منفذ أكبر من 1024 بالولوج داخل الشبكة الداخلية، وفي هذا نوع مخاطرة ولهذا يجب أن نضيف طبقة أمنية أخرى، إذ يمكن أن نغير هذه القاعدة لاستقبال أجوبة خوادم الويب فقط، وذلك بجعل منفذ المصدر رقم 80 وأيضًا لا بد أن يكون الطرد إجابة على طلبية أرسلت قبل ذلك، ويكون هذا بالتأكد أن بت التأكيد ack مفعلة.

جدول 4. 2 خيارات متقدمة في قواعد التصفية.

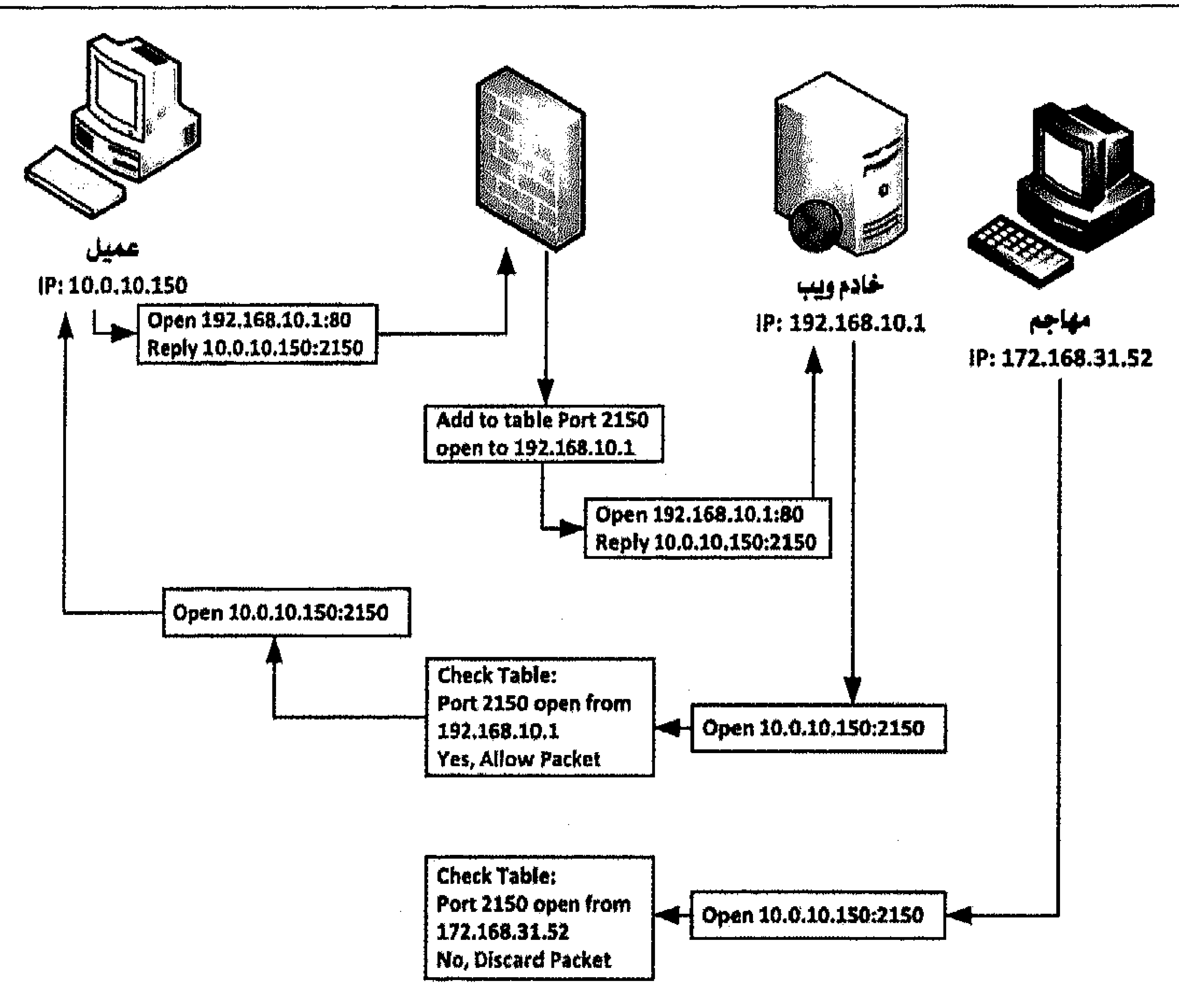
Rule	Direction	Protocol	Src. Address	Dest. Address	Src. Port	Dest. Port	ACK	Action
4	Inbound	TCP	External	Internal	80	>=1024	Set	Allow

1. 3. 2 طرق التصفية

هناك نوعان من المصفيات، وهما المصفيات التي لا تعمل بتقنية التذكر، والمصفيات ذات الذاكرة. أما المصفيات غير ذات ذاكرة فإنها تتعامل مع كل طرد على أنه وحدة منفصلة عن الاتصال تمامًا، إذ إنها لا تحتفظ بتاتًا بأي معلومات على الاتصالات التي تمر بها. بل أنها تتخذ القرار لكل طرد على حدة، بناء على عنوانه ومنفذه، وبعض المعلومات حسب نوع البرتوكول المستعمل.

والنوع الثاني من المصفيات هو الذي يحتفظ بمعلومات على الاتصالات والجلسات، ولذلك يمكنه أن يتعرف على أي تطفل على اتصال أو جلسة، لأنه ينشئ قائمة عناوين المتصلين ببعض، فلا يسمح بالولوج إلا لطرود طرفي الاتصال. وبهذا فإن مستوى الأمن الذي تقدمه هذه المصفيات أعلى، لأنها لا تعتمد في التصفية على معلومات الطرود كل على حده، بل أيضًا على إطار الاتصال والجلسة ككل.

شكل 4.4 عمل المصفيات بتقنية التذكر.



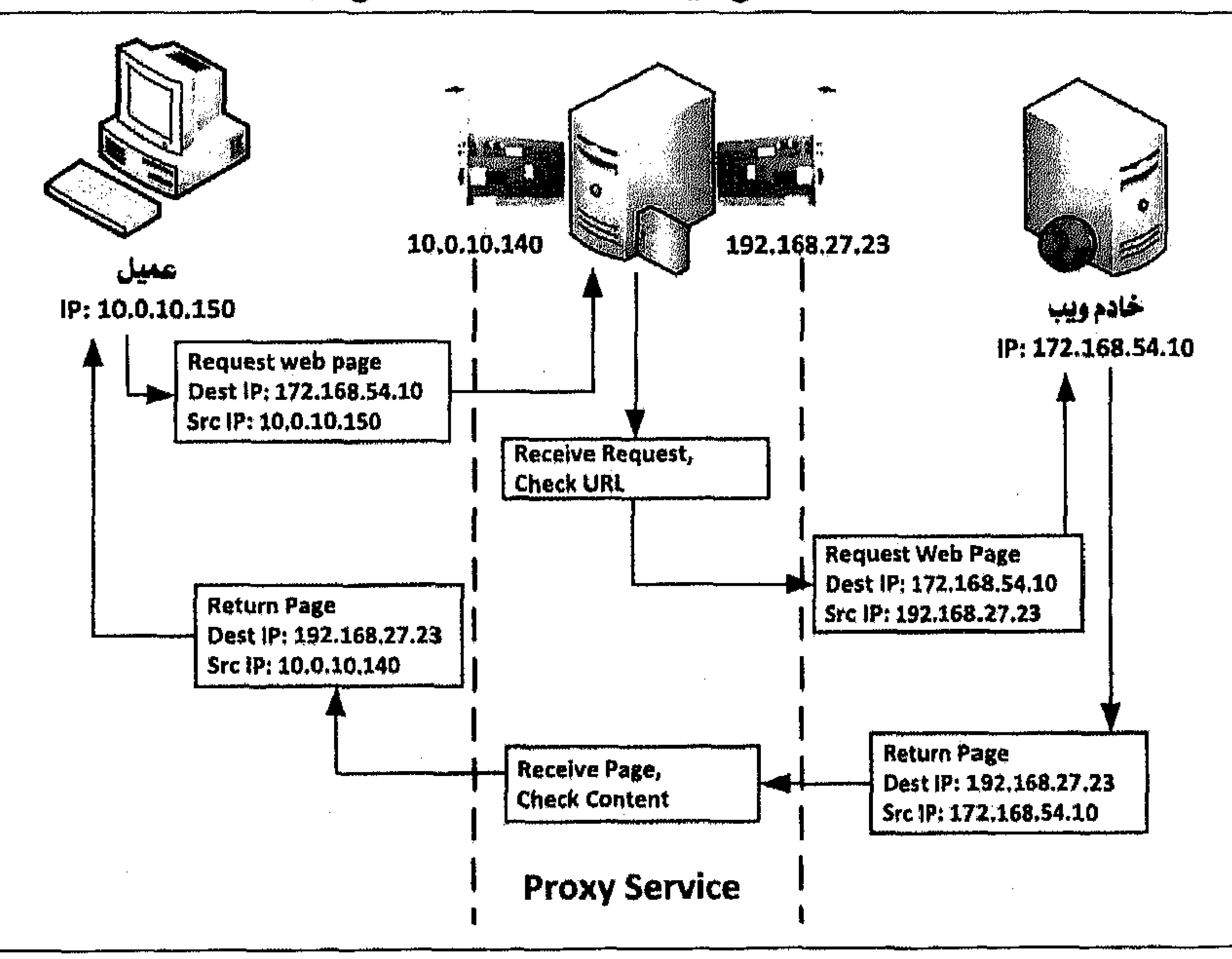
يعرض الشكل مثالاً لطريقة عمل المصفيات التي تعمل بتقنية التذكر. عندما يتصل العميل بمقبسه(10.0.10.150، 2150) على مقبس خادم الويب (192.168.10.1، 80) يمر الطرد بمصفي الطرود، فيحتفظ بمعلومة الاتصال إلى قائمة الاتصالات، وهي أن المنفذ رقم 2150 ينتظر إجابة من العنوان 192.168.10.1 ثم يمرر طلبية العميل إلى الخادم فلما يجيب

الخادم يتأكد جدار الحماية من أن الطرد قادم من عنوان الخادم وهو 192.168.10.1 فإن كان كذلك فيسمح له بالولوج وإلا فلا. فلو أن مهاجمًا 172.168.31.52 أراد أن ينتحل شخصية الخادم فإن جدار الحماية لا يسمح له بالولوج؛ لأن عنوانه غير موجود في قائمة عناوين الاتصالات. ولو أن المصفي لا يعمل بتقنية التذكر لسمح له بالدخول.

4. 2 الخادم الوكيل أو البوابة التطبيقية

يعمل الخادم الوكيل أو البوابة التطبيقية كوكيل عن كل أجهزة الشبكة الداخلية، فمثلا عندما يتصل العميل 10.0.10.150 بخادم الويب 192.168.54.1. يمر الطرد بالخادم الوكيل فيفتش محتواه، فإن كان مسموحًا به يقوم الوكيل بإنشاء طردٍ آخر يضع فيه عنوانه عِوَضَ عنوان العميل، فيجيب خادم الويب على الخادم الوكيل عوض أن يجيب على العميل، ولما تصل الإجابة للخادم الوكيل يقوم بتفتيش محتواه، فإن كان مسموحًا به يقوم بإرساله إلى العميل.

شكل 4. 5 توصيف عمل الخادم الوكيل.



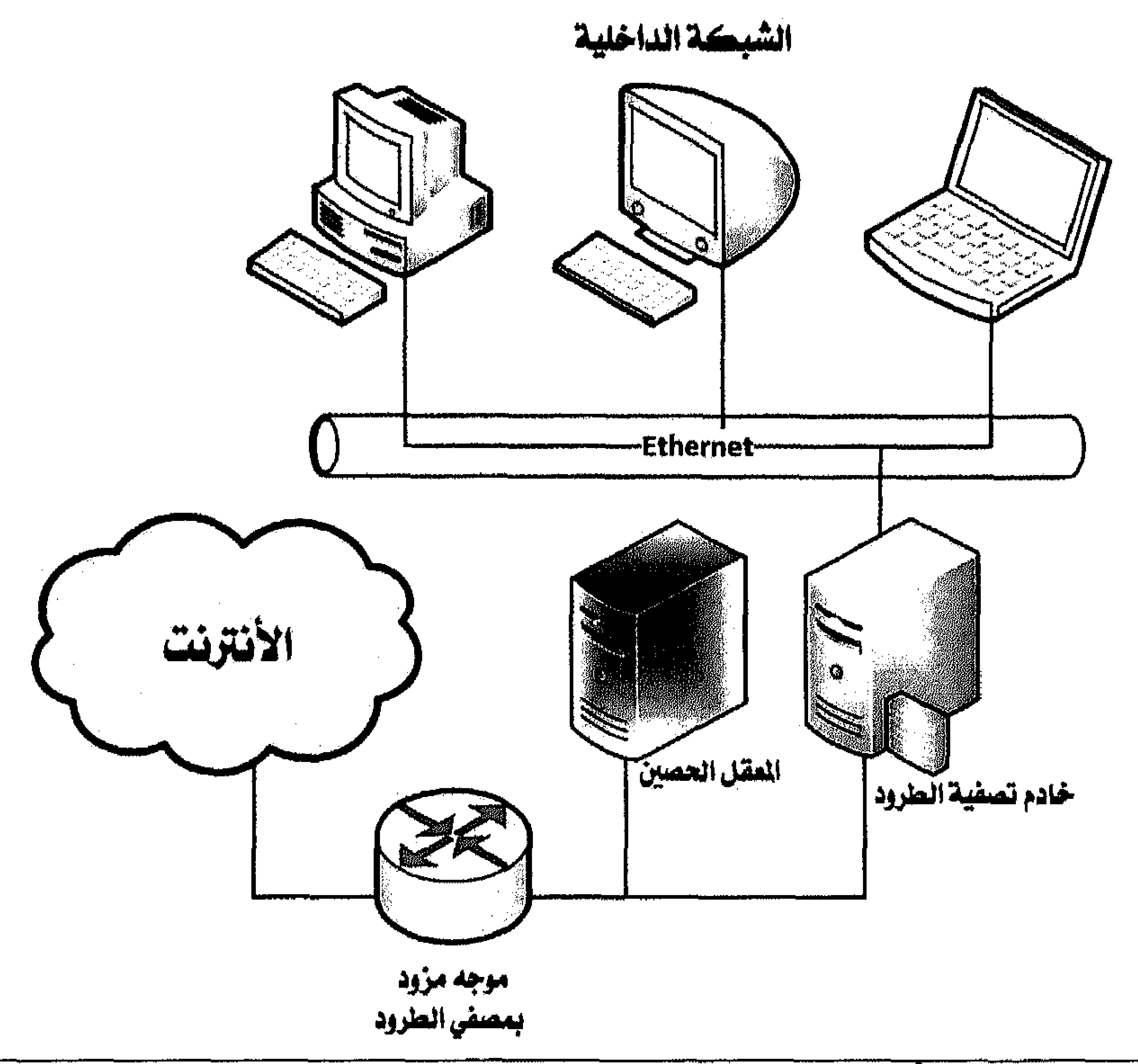
للخادم الوكيل عدة فوائد. أولها: أنه يخفي كل عناوين الشبكة الداخلية، فلا تكون عرضة للهجومات، لأنها مجهولة. ثانيها: أنه يقوم بالتصفية بالنظر إلى محتوى الطرود لا عناوينها فقط كما في مصفي الطرود. ثالثها: أنه يوفر لك سجلاً تتبع واحدا لكل الأجهزة، مما يسهل عملية مراقبة الشبكة، وكل أنشطة المستخدمين ويوفر الوقت. ولكن المشكل الأساس للخادم الوكيل أنه يمثل نقطة هشة في الشبكة، فلو تعطل لتعطلت كل الشبكة، ولذلك يتطلب مجهود حماية أكثر من غيره. كما أنه يكون عرضة للازدحام، لأنه المنفذ الوحيد لجميع الأجهزة الداخلية، مما قد يسبب بطئاً شديداً في الشبكة. كحل لهذه المشاكل يمكن أن نجعل لكل خدمة خادماً وكيلاً، ولكن هذا الحل – وإن كان يحد من عيوب الخادم الوكيل – إلا أنه يعقد عملية إدارة الشبكة، ويستهلك الوقت والموارد ولهذا نقتصر عادة على وضع خوادم وكيلة لأكثر خدمات الإنترنت استعمالاً كخدمة البريد والويب. يجدر الملاحظة أيضاً أنه يجب الاعتناء بملفات ضبط خوادم الوكيل، والقيام بتخصيصها، وإلا فإن المهاجمين يمكن أن يعتمدوا على الخصائص الافتراضية للخوادم ومن ثم مهاجمتها.

5. 2 المعقل الحصين وجدار الحماية

المعقل الحصين هو عبارة عن جهاز يقع تحصينه بشكل أكبر من أي جهاز آخر على الشبكة، وذلك بالاستفادة القصوى من كل خيار أمني مدمج في نظام التشغيل، وتجريده من كل حسابات المستخدمين، والخدمات والتطبيقات والإبقاء فقط على الأشياء التي لا بد منها وما منها بد.

يمكن أن نجعل المعقل الحصين في أي مكان من الشبكة، ولكن عادة ما نجعله في المنطقة الخضراء، ويمكن حينها أن نجعل في المعقل أي خدمة من خدمات الانترنت كخدمة الويب مثلا.

شكل 6.4 مكان وضع المعقل الحصين.

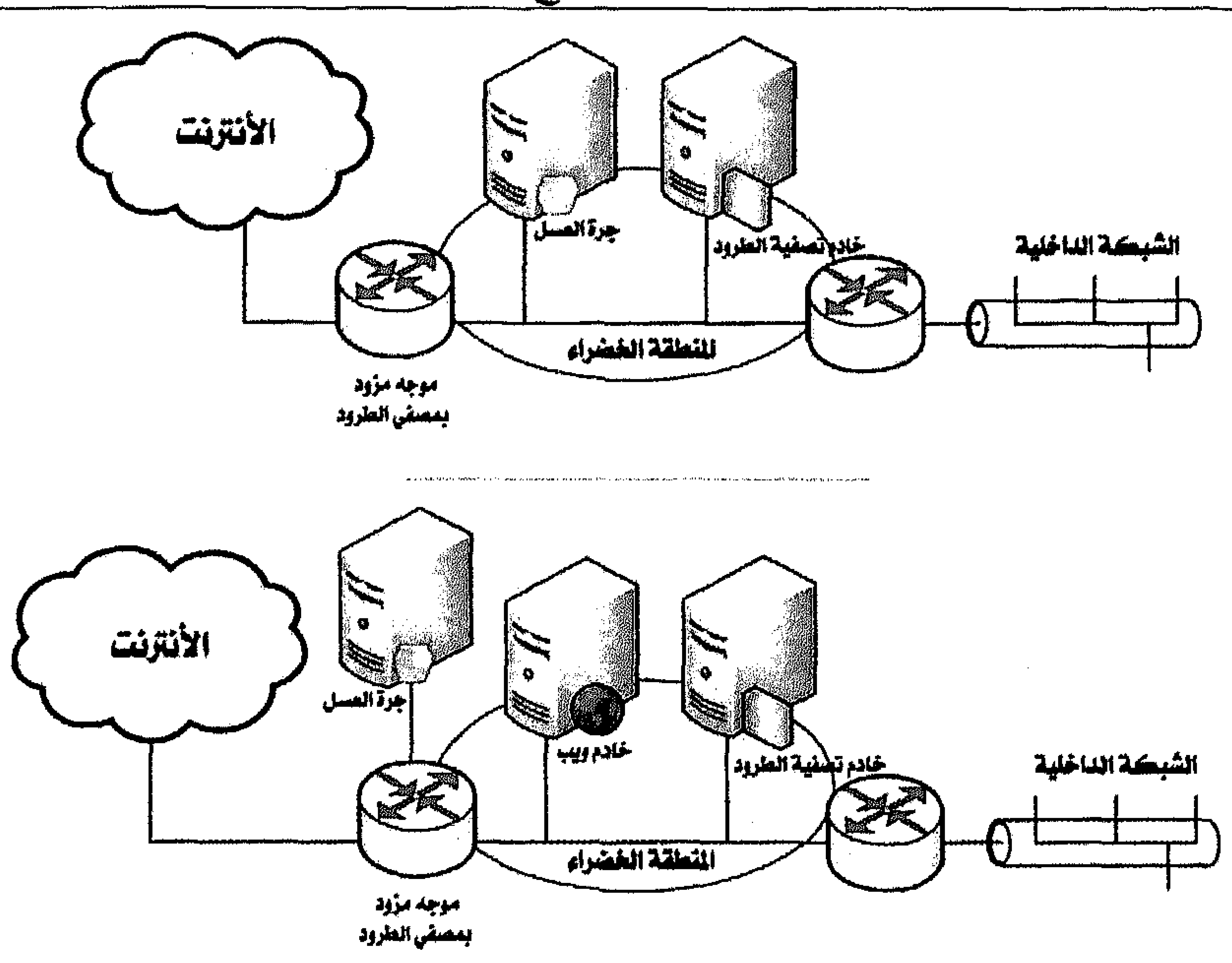


كما يمكن أن يستعمل المعقل الحصين للقيام بمهام جدار الحماية، فتتم عملية تثبيت نظام التشغيل من البداية بعد تهيئة القرص الصلب دون إعطاء خيار إمكانية الدخول المتعدد على أكثر من نظام تشغيل. ثم ينزع كل عتاد مادي غير مستعمل، كالموديم والبطاقات الصوتية. كما لا بد من استعمال أعلى درجات التأكد من الهوية، ومن المفضل أن تكون باستعمال الخصائص البيولوجية كما يجب وضع برنامج التأكد من سلامة البيانات والملفات، لاكتشاف أي تغيير يحدث فيها، ومن ذلك برنامج TripWire مثلا. وعندما يتعرض المعقل الحصين إلى الهجوم فيجب تهيئته من جديد، وإعادة تثبيته لضمان أن المهاجم لا يعرف أية معلومة قديمة عليه من شأنها أن تعرضه من جديد إلى هجوم آخر.

2.6 جرة العسل وجدار الحماية

يهدف جهاز جرة العسل أو الشرك المغري إلى استقطاب المهاجمين بقصد متابعة أنشطتهم، والتعرف عليهم وعلى نواياهم وطرقهم في الهجوم. كما أنها تستنفد مجهودات المهاجمين بحيث يكون تركيز المهاجمين على الشرك لا على الشبكة الحقيقية. وككل شرك لا بد أن يكون مخفيًا ومغريًا بالقدر اللازم، ولهذا فإن تصميم جرة العسل لا يعتمد على مهارات تقنية فحسب، بل أيضًا نفسية كمعرفة سيكولوجية المهاجم، وما الذي يغري والذي لا يغري. فمثلا لا تكون الثغرات في جرة العسل بديهية بل لا بد من تعقيدها القدر الكافي، لكي لا يكتشف المهاجم أنه في شرك. كما أنه من غير المفترض أن تكون جرة العسل معقلاً حصينًا، لأننا بحاجة للتعرف على طرق الهجوم الواقعة على جرة العسل للاستفادة منها، وذلك عن طريق أخذ الحيطة، وتحصين الشبكة الحقيقية منها قبل وقوعها. كما أنه من أكبر مزايا جرة العسل أنها ترفع القدرة على اكتشاف الاختراقات، وتسريع أساليب الدفاع على الهجومات. كما يمكن أن يتطور الشرك من جهاز واحد ليصبح شبكة كاملة تعرف بشبكة العسل، أو الشبكة المفخخة، وتحتوي على أجهزة وخدمات مشابهة للشبكة الحقيقية، وفي هذه الحالة تكون الاستفادة أكثر والتصميم أعقد نوعًا ما.

شكل 7.4 مكان وضع جرة العسل.



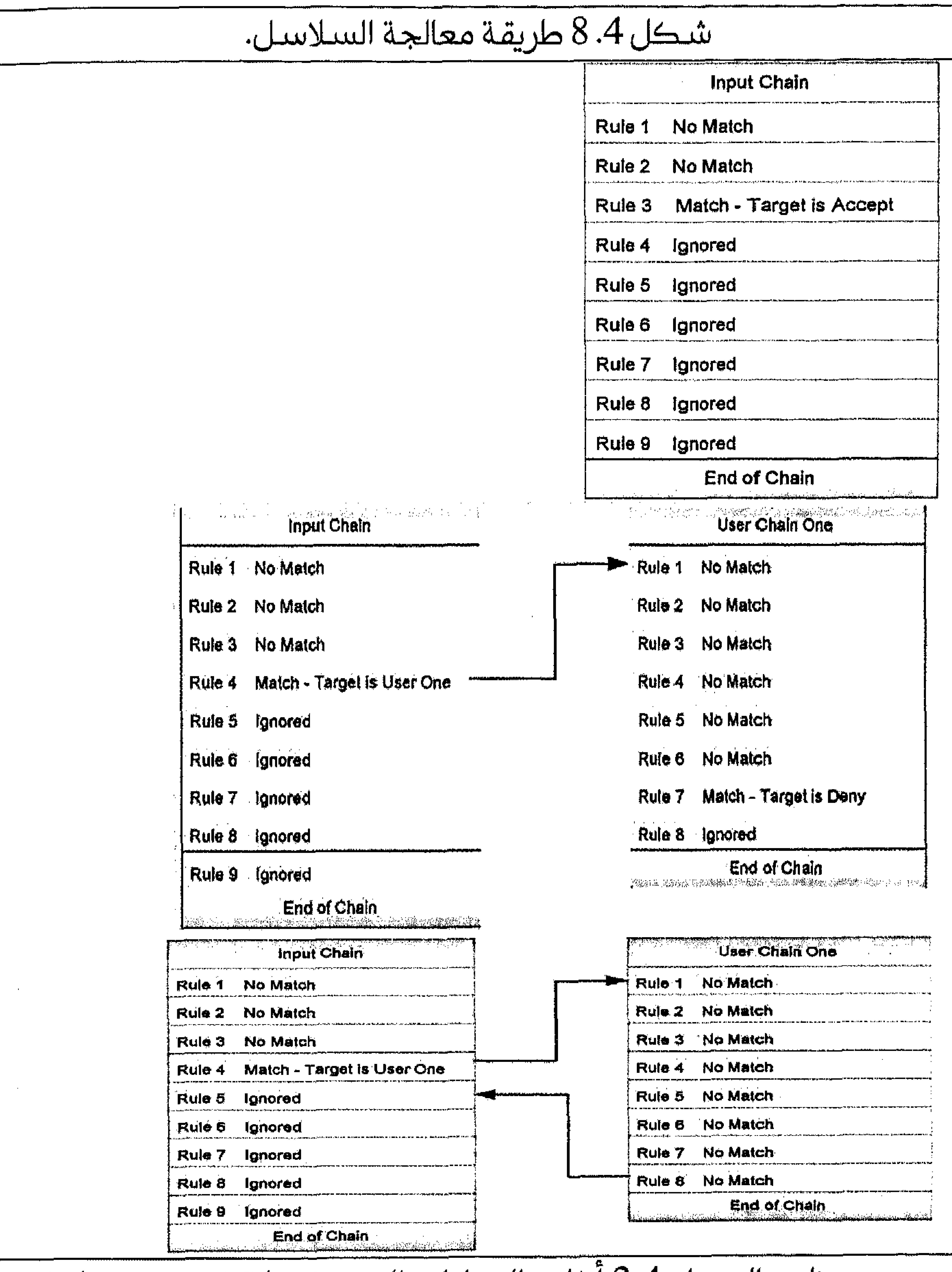
يمكن وضع جرة العسل داخل المنطقة الخضراء إذا أردنا أن نتابع تحركات المهاجمين الذين استطاعوا تجاوز مصفي الطرود الأول، وأحيانًا يمكن أن نسمح عمدًا بولوج بعض الطرود الممنوعة، وتوجيهها إلى جرة العسل. إما إن كان الغرض معرفة الهجومات على مصفي الطرود نفسه، فنشبك جرة العسل مباشرة به فيكون طرفًا ثالثًا للموجه، ومن ثم نحاول التعرف على الهجومات التي تستهدف جدار الحماية نفسه. أخيرًا لا بد من الإشارة إلى أنه لا بد من الأخذ بعين الاعتبار المسائل القانونية التي تنجر عن تصميم، وتنفيذ جرة العسل أو الشبكة المفخخة؛ لأن المستخدم العادي الذي لا يهدف إلى الهجوم على الشبكة يمكن أن يدلي بمعلومات خاصة أو لا يسمح أصلاً بتتبع تحركاته على الإنترنت، ويعتبر ذلك انتهاكًا لخصوصيته، ومن ثم يمكن أن ترفع الدعاوى القضائية على المؤسسة في حالة اكتشاف المستخدم ذاك، وعمومًا لا بد من الرجوع للتشريعات القانونية الخاصة بخدمات الإنترنت.

2.7 ضبط جدار الحماية IPTABLES

نعرض في الفقرات الموالية لجدار حماية IPtables الذي يعتمد على مصفي طرود يمكن تثبيته مع نظام التشغيل، أو إضافته لاحقًا عليه. ويعمل هذا البرنامج على نظامي يونكس ولينكس، ويتكوم من ثلاثة قوائم، وهي: قائمة التصفية وقائمة محول عناوين الإنترنت، وقائمة تغيير حقول الرؤوس الطرود. وسنعرض أساسًا في الفقرات الموالية إلى تصفية الطرود عن طريق قائمة التصفية. يتعامل الجدار مع ثلاث سلاسل مدمجة لا يمكن مسحها من النظام، وهي سلسلة الداخل والخارج والمعاد إرساله (INPUT, OUTPOUT, FORWARD) وعندما يحدث تطابق من حزم البيانات الداخلة أو الخارجة أو المعاد إرسالها مع قاعدة من قواعد التصفية في هذه السلاسل يقوم المصفي سياسة من السياسات الثلاث الموالية: إما بالسماح لها بالدخول(ACCEPT)، أو نبذها وحذفها بدون إشعار المرسل(DROP)، أو رفضها مع إشعار المرسل بالرفض (REJECT). وفي كل الحالات إذا لم يوجد تطابق للطرد مع إحدى القواعد، فإن القاعدة الافتراضية للجميع هي التي يقع تطبيقها.

يتم البحث عن التطابق في السلاسل بشكل تسلسلي، وعند أول تطابق يقع تنفيذ السياسة المحددة في القاعدة، فمثلا في المثال وجد تطابق مع القاعدة الثالثة والسياسة هي قبول الطرد، كما يمكن أن يكون القرار عند التطابق مع قاعدة معينة أن يتم تحويل البحث عن التطابق في قائمة ثانية وحيثما وجد التطابق يطبق القرار ففي المثال وجد تطابق عند القاعدة الرابعة، وتم تحويل البحث في قائمة ثانية، حيث وجد تطابق مع القاعدة السابعة، وكان القرار النهائي رفض الطرد. وفي بعض الحالات عندما لا يوجد تطابق في القائمة الثانية يتم الرجوع إلى القائمة الأولى لإتمام عملية البحث عن التطابق مع القواعد الباقية في السلسلة.

شكل 8.4 طريقة معالجة السلاسل.



يظهر الجدول 3.4 أغلب العمليات التي يوفرها IPTABELS لإدارة السلاسل والقواعد، وإنشاء القواعد وبقية الخيارات الأخرى.

جدول 4. 3 أوامر التحكم في Iptables

إدارة السلاسل	
إنشاء سلسلة جديدة	Iptables -N chainname
حذف سلسلة	Iptables -X chainname
إضافة سياسة للسلسلة	Iptables -P input DROP
سرد قواعد السلسلة	Iptables -L chainname
مسح كل قواعد السلسلة	Iptables -F chainname
إدارة قواعد التصفية	
إضافة قاعدة في آخر السلسلة	Iptables -A chainname rule
إضافة قاعدة في موضع معين بالسلسلة	Iptables -I chainname rule-number rule
استبدال قاعدة باعتبار رقمها بأخرى	Iptables -R chainname rule-number rule
حذف قاعدة باعتبار رقمها	Iptables -D chainname rule-number
حذف قاعدة بكتبة محتواها	Iptables -D chainname rule
إنشاء قواعد التصفية	
تحديد مصادر الطرود (عنوان انترنت)	Iptables -s source [/mask]
تحديد وجهة الطرود (عنوان انترنت)	Iptables -s source [/mask]
تحديد البرتوكول المستعمل	Iptables -p protocol
الذهاب إلى السلسلة المحددة دون عودة	Iptables -g chainname
الذهاب إلى القرار مثل القبول أو الرفض أو الذهاب إلى سلسلة أخرى	Iptables -j target
تحديد طرود SYN في اتصال TCP	Iptables --syn

خيارات أخرى	
تحديد منفذ المرسل إليه	Iptables --dport
تحديد منفذ المرسل	Iptables --sport
تحديد مجال من المنافذ	Range of ports 1:1024
استعمال عامل "ماعدا"	!entry
استعمال رمز يعني أي عنوان انترنت	0/0 or any

2.7.1 شرح أمثلة من قواعد جدار الحماية IPTABLES

```
iptables -F chain5
```

- مسح جميع قواعد السلسلة chain5.

```
iptables -A chain7 -p TCP -s 10.0.10.11 - syn
```

- إضافة قاعدة للسلسلة chain7، وهي خاصة بطرود بروتوكولTCP المرسلة من العنوان 10.0.10.11 والخاصة بإنشاء اتصال TCP.

```
iptables -A output -p TCP -d ! 172.168.33.44 --dport 80
```

- إضافة قاعدة للسلسلة output خاصة بطرود بروتوكول TCP المتجهة إلى أي خادم ويب في الانترنت ما عدا الخادم 172.168.33.44

```
iptables -A output -p TCP -d  172.168.33.44 --dport ! 80
```

- إضافة قاعدة للسلسلة output خاصة بطرود بروتوكول TCP المتجهة إلى أي خدمة انترنت على الخادم 172.168.33.44 ما عدا خدمة الويب

```
iptables -A input -s 10.0.10.100 -j DROP
```

- إضافة قاعدة للسلسلة input لحذف كل الطرود المرسلة من 10.0.10.100

```
iptables -A input -p TCP -d 0.0.0.0/0 12345 -j DROP
```

- إضافة قاعدة للسلسلة input لحذف كل طرود TCP الموجهة إلى أي عنوان انترنت على منفذ 12345 وهو منفذ لبرنامج حصان طروادة معروف

```
iptables -A output -p TCP -s 10.0.10.0/24 -d 0.0.0.0/0 -- dport
80 -j ACCEPT
```

- إضافة قاعدة للسلسلة output لقبول كل طرود TCP المرسلة من الشبكة 10.0.10.0/24 والموجهة إلى أي عنوان في الإنترنت يشغل خادم ويب على منفذ رقم 80

```
iptables -A output -p TCP -s 0.0.0.0/0 -d 10.0.10.0/24 -- dport
31337 -j DROP
```

- إضافة قاعدة للسلسلة output لحذف طرود TCP المرسلة من أي عنوان والموجهة لعناوين الشبكة 10.0.10.0/24 على المنفذ 31337

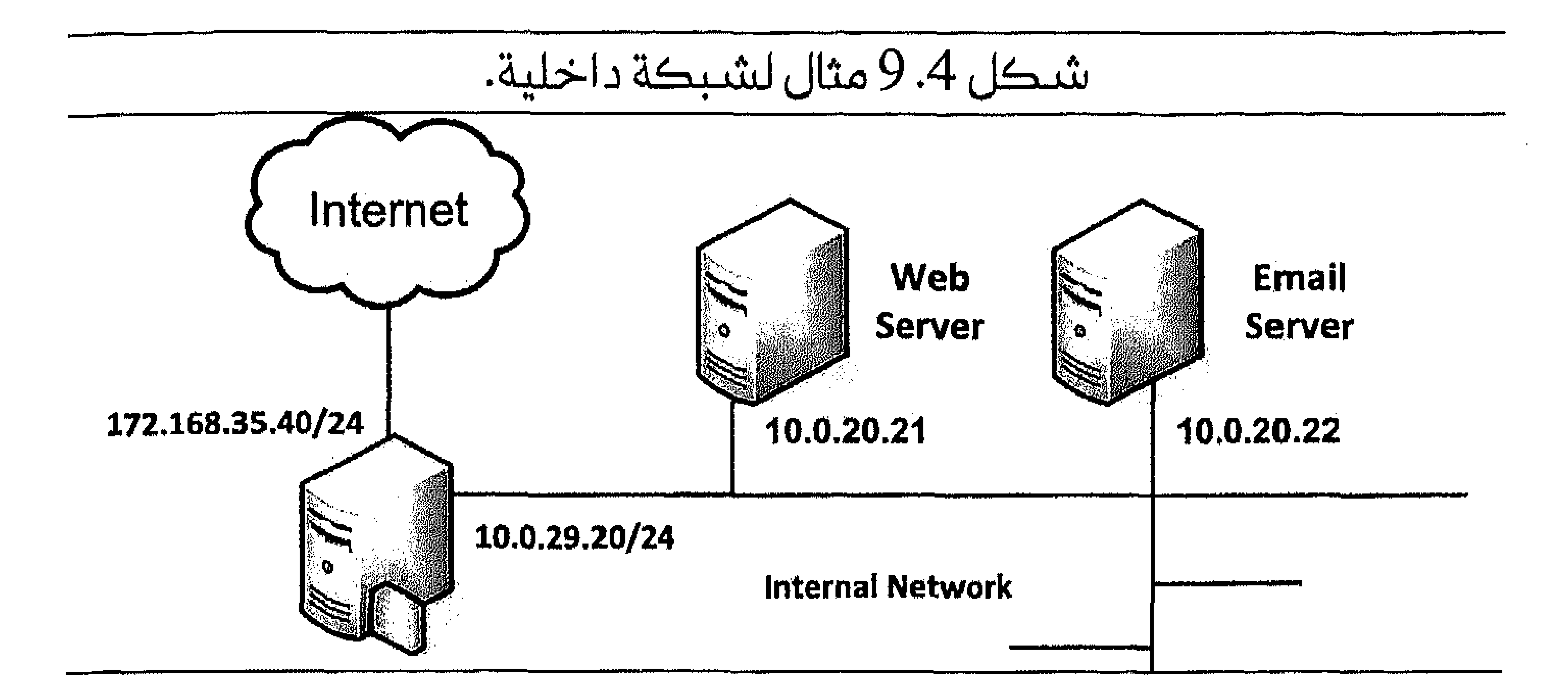
```
iptables -A output -p TCP -s 0.0.0.0/0 -d 10.0.10.0/24 --
dport 5000:10000 -j DROP
```

- إضافة قاعدة للسلسلة output لحذف طرود TCP المرسلة من أي عنوان والموجهة لعناوين الشبكة 10.0.10.0/24 على المنافذ 5000 إلى 10000

2 .7 .2 دراسة حالة

- في هذه الفقرة نحاول تطبيق ما درسناه على حالة واقعية، فنبدأ بتحديد أهداف وسياسة جدار الحماية، ومن ثم صناعة السلاسل وقواعد التصفية اللازمة لذلك. لنفترض البنية التحتية المعروضة في الشكل 4. 9. هدفنا هو التحكم في حزم البيانات من وإلى شبكة الإنترنت والشبكة الداخلية. يوجد في الشبكة الداخلية مصفي طرود وخادم ويب وبريد.

شكل 4. 9 مثال لشبكة داخلية.



- ولتكن أهداف جدار الحماية التالية:

1. السماح لحزم بيانات برتوكول ICMP للعملاء داخل شبكة المؤسسة.
2. السماح لعملاء المؤسسة الخارجيين من الوصول لخادم البريد.
3. عدم السماح للعملاء من داخل المؤسسة بالاتصال بخوادم بريدية على الإنترنت.
4. السماح للمستخدمين الخارجيين بالوصول لخادم ويب المؤسسة.
5. إيقاف هجومات الخداع بإدراج عناوين بريدية مختلفة IP SPOOFING

عادة ما يقوم الضبط الافتراضي لمصفي الطرود على حذف كل طرد يتطابق مع ثلاث سلاسل مدمجة:

```
iptables -P INPUT -j DROP
iptables -P OUTPUT -j DROP
iptables -P FORWARD -j DROP
```

1. إنشاء سلسلتين جديدتين، واحدة خاصة بالطرود الداخلة للشبكة الداخلية them-us وواحدة خاصة بالطرود الخارجة منها us-them.

```
iptables -N us-them
iptables -N them-us
```

2. قاعدتان للسماح للطرود الخارجة من الشبكة بالاتصال بكل عنوان غير عناوين الشبكة الداخلية وكذا بالنسبة للطرود الداخلة للشبكة.

```
iptables -A INPUT -s 10.0.20.0/24 -d ! 10.0.20.0/24 -j us-them
iptables -A INPUT -s ! 10.0.20.0/24 -d  10.0.20.0/24 -j them-us
```

3. قاعدتان لتمكين العملاء داخل الشبكة الداخلية من الوصول إلى أي خادم ويب خارجي والتواصل معه.

```
iptables -A us-them -p TCP -d 0/0 - dport 80 -j ACCEPT
iptables -A us-them -p ICMP -d 0/0 - dport 80 -j ACCEPT
```

4. قواعد لتمكين المستخدمين من الخارج من الوصول إلى خادم البريد والويب، وأما منع IPSPOOFING من داخل الشبكة الداخلية فعبرت عنها القاعدة الأخيرة.

```
iptables -A them-us -p TCP -d 10.0.20.22 - dport 25 -j ACCEPT
iptables -A them-us -p TCP -d 10.0.20.22 - dport 110 -j ACCEPT
iptables -A them-us -p TCP -d 10.0.20.21 - dport 80 -j ACCEPT
iptables -A them-us -s 10.0.20.0/24 -j DROP
```

3 - نظم اكتشاف الاختراقات

يقوم نظام اكتشاف الاختراقات (IDS (Intrusion Detection System بكشف الأنشطة الضارة في جهاز الحاسب والشبكة. فهو يتعرف ويوقف الهجومات في وقتها، كما يقوم بالتحقيق في آثار الجريمة عند نجاح هجوم عدواني على الشبكة. يتكاتف ويتكامل عمل نظام اكتشاف الاختراقات مع عمل الوسائل الوقائية، كجدران الحماية إذ يقوم جدار الحماية بعملية تصفية الطرود بناء على سياسة أمنية وقواعد محددة، ولكنه لا يقوم بالتحقق من أن الطرود التي يسمح لها بالولوج للشبكة لا تحمل هجوما ما. يكمل نظام اكتشاف الاختراقات عمل جدار الحماية بالتحقق من سلامة الطرود المسموح لها بالولوج من كل هجوم ضار.

عندما يحكم نظام اكتشاف الاختراقات على طرد معين بأنه يحمل هجوما، أولاً: فإن حكمه يمكن أن يكون صحيحًا أو خطأ، وكذلك في حالة ما إذا حكم بسلامة الطرد من هجوم ما، فإن هذا الحكم يمكن أن يكون صوابًا أو خطأ فتحصل عندنا أربع حالات:

1. إن حكم بأنه لا يوجد هجوم، وفي الحقيقية لا يوجد هجوم فعلا فالحكم صواب ويعبر عنه ب TN (True Negative)
2. إن حكم بأنه يوجد هجوم، وفي الحقيقية هناك هجوم فعلا فالحكم صواب ويعبر عنه ب TP (True Positive)
3. إن حكم بأنه يوجد هجوم، وفي الحقيقية لا يوجد هجوم فعلا فالحكم خطأ ويعبر عنه ب FP (False Positive)
4. إن حكم بأنه لا يوجد هجوم، وفي الحقيقية هناك هجوم فعلا فالحكم خطأ ويعبر عنه ب FN (False Negative)

فالحالتان الأولتان لا إشكال فيهما، وأما الحالة الثلاثة فإن فيها استهلاكًا للموارد من تخزين لتحذيرات لا حقيقة لها، وتستهلك وقت مدير الشبكة في متابعة لشيء لا وجود له. أخطر الحالات هي الحالة الرابعة حيث يسمح النظام للمهاجم من الوصول لبغيته وذلك بدون إشعار مدير الشبكة بذلك، بل يشعره أن الوضع آمن وليس هناك هجوم أصلاً. تسعى نظم اكتشاف الاختراقات من التقليل من FP ومن FN قدر الإمكان باستعمال أكثر من طريقة في اكتشاف الاختراقات، واستعمال أكثر من نوع من نظم اكتشاف الاختراقات.

1. 1. 3 أنواع نظم اكتشاف الاختراقات

يمكن تقسيم نظم اكتشاف الاختراقات إلى عدة أنواع، بحسب مكان عملها أو طريقة وأسلوب الاكتشاف لديها، أو بحسب تعاملها مع البيانات التي تمسحها وردها على الهجوم المكتشف.
هناك ثلاثة أنواع رئيسة في نظم اكتشاف الاختراقات بحسب مكان عملها:

1. **نظام يعمل على مضيف:** يعمل هذا النظام على حاسوب واحد ويقوم بمسح البيانات الداخلة والخارجة منه بهدف اكتشاف الاختراقات، وله القدرة على إيقاف تقدم هجوم ما تم اكتشافه. بداية استعمل هذا النظام لحماية الخوادم المهمة في الشبكة حيث يقوم بتحليل سجلات لنظم التشغيل والتطبيقات واستعمال الموارد، وبقية أنشطة النظام الموجودة على جهاز

الخادم. يقوم النظام اكتشاف الاختراقات التي تحدث على المضيف بمراجعة مختلف أنواع السجلات ومراقبة التغييرات التي تحصل على مختلف الملفات بحساب بصماتها، ومتابعة الحركة التي تحصل على المنافذ، ومراقبة الطلبيات قبل معالجتها، ومراقبة أنشطة العمليات Processes على الحاسوب. إذا ما اكتشف النظام عملاً هجوميًا ونشاطًا مشكوكًا فيه، فإنه يقوم بتسجيل الواقعة، ويشعر مدير النظام بذلك ثم يقوم بإخراج المستخدم من جلسته ويوقف حسابه.

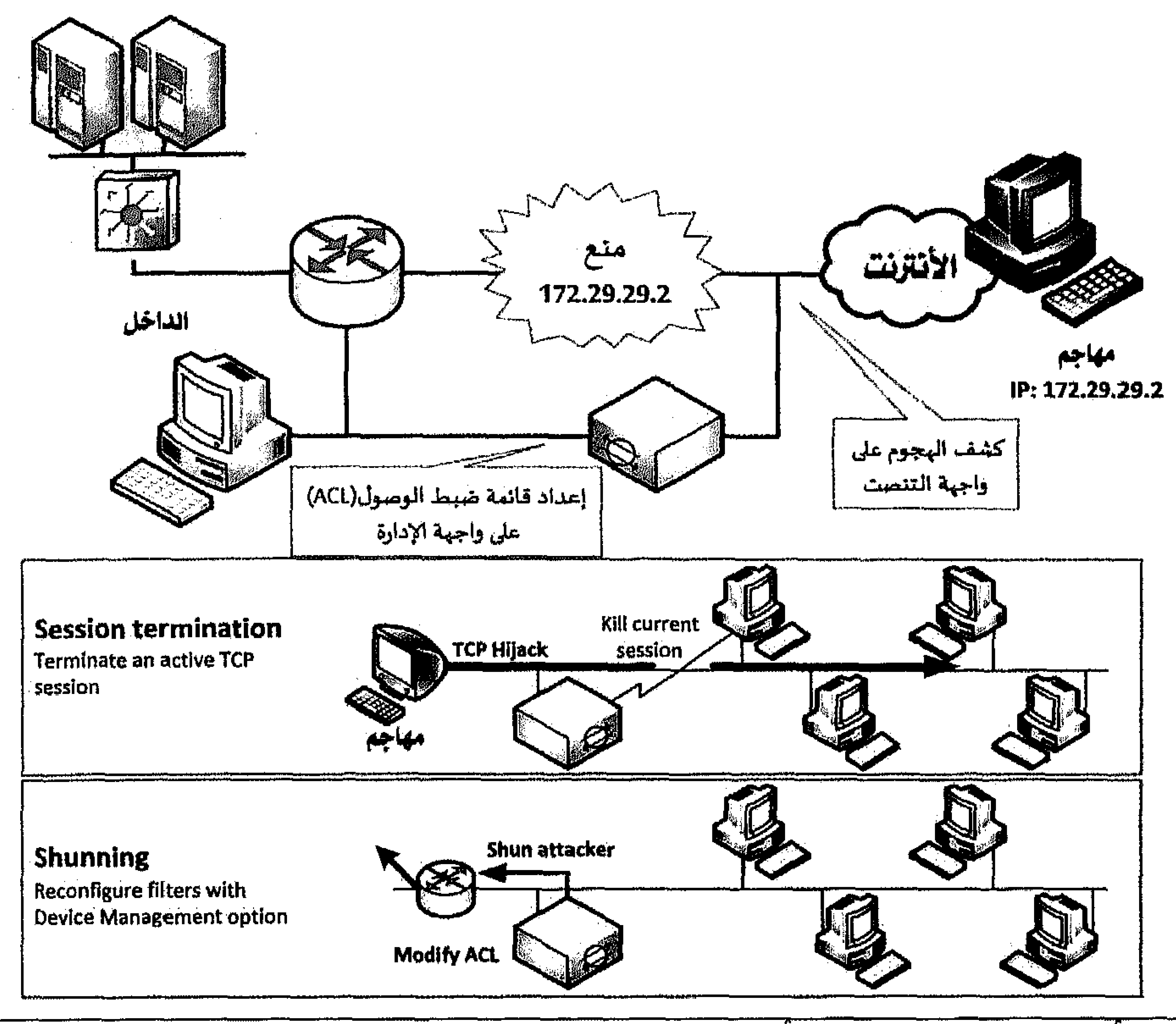
2. **نظام يعمل على الشبكة:** يقوم هذا النظام بمسح البيانات الداخلة إلى الشبكة والخارجة منها لاكتشاف الاختراقات، والتنبيه المسبق على وجود هجوم ما قبل أن يصل إلى النظام المستهدف بالهجوم. يقع وضع هذا النظام إما داخل جدار الحماية أو في المنطقة الخضراء DMZ أو في قطعة الشبكة التي تكون فيها الخوادم والموارد المراد حمايتها.
3. **نظام موزع:** يضيف هذا النظام لنظام اكتشاف الاختراقات على الشبكة بعد الإدارة الفعالة له، والتعامل الصحيح مع الهجوم في أسرع وقت ممكن.

كما يمكن أن نقسم نظم اكتشاف الاختراقات على حسب طريقة تعاملها مع البيانات الداخلة للشبكة وردة فعلها على هجوم مكتشف. هناك نوعان أساسيان وهما:

1. **نظم الاكتشاف الخاملة:** وهي التي تشمم البيانات من الخطوط الشبكية دون أن تمنع مرورها، ثم تقوم بمسحها فإن اكتشفت هجومًا ما فإنها تقوم بتسجيل الحالة وإشعار المدير، ولكن لا تقوم بأي شيء يمنع دخول الطرد الحامل للهجوم، ولا تمنع الهجوم من التقدم.

2. **نظم الاكتشاف النشطة:** تقوم بنفس عمليات النظم الخاملة، ولكنها تتميز عنها بقدرتها على القيام بعمليات دفاعية لتجنب الهجوم أو صده (IDS Shunning or blocking)، أو إيقاف الاتصال نهائيًا (TCP reset) (انظر الشكل 10.4). وهذا النظام يستعمل في الشبكة عند قيام المدير بضبط عمل النظام وتقليل عدد إنذارات FP.

شكل 10.4 مثال عمليات دفاعية لتجنب الهجوم أو صده.



أما من حيث طريقة وأسلوب الاكتشاف لدى نظم اكتشاف الاختراقات فإنها تنقسم إلى ثلاثة أقسام وهي:

1. النظم التي تراقب سلامة الطرود (Misuse IDS or Signature based IDS): هذه النظم شبيهة جدًا بعمل البرامج المضادة للفيروسات، إذ تمتلك قائمة من أنماط الهجومات المعروفة، وفي كل مرة تقوم بمسح الطرود للبحث عن هذه الأنماط فيها، فإن حصلت توافقًا مع نمط معين فإنها تصدر إنذارًا بذلك، وتعتبر أن هذا الطرد يحمل هجومًا. توضع هذه الأنماط في شكل قواعد مضادة للتهديدات التي يقع اكتشافها فلا بد من تحديثها دومًا لكي يكون النظام فعالاً. من أكبر سلبيات هذا النظام عدم قدرته على

اكتشاف الاختراقات التي لم تعرف بعد، والحاجة الدائمة إلى تحديث قواعد المسح كلما اكتشف تهديد جديد.

2. النظم التي تراقب سلوك الشبكة(Anomaly based IDS): تعتمد هذه النظم على مقارنة حالة الشبكة، وحركة البيانات فيها في كل فترة زمنية معينة مع الحالة المعيارية، التي يقع ضبطها على أنها الحالة الطبيعية للشبكة. فكل اختلاف مع هذه الحالة المعيارية يعني أن هناك شيئًا غير طبيعي يحدث في الشبكة، فتصدر هذه الأنظمة تحذيرًا بإمكانية هجوم لمدير الشبكة الذي من مهمته التأكد من ذلك. تمتاز هذه الأنظمة على سابقتها بأنها بإمكانها اكتشاف الهجومات في المستقبل، ولا تحتاج إلى تحديث، ولكن الصعوبة فيها هو في كيفية تحديد النشاط الطبيعي للشبكة، والحكم بأن هناك نشاطًا غير طبيعي فيها. ينبني على تحديد هذه العوامل عدد الإنذارات التي يمكن أن تغرق مدير الشبكة، وتستهلك وقته إذا اعتبرنا أن كل تغيير – ولو طفيفًا – يصدر إنذارًا. وبالعكس إذا لم تضبط هذه العوامل بشكل جيد، واعتبرنا أن لا هجوم مشتبه إلا إذا لوحظ تغيير كبير بسلوك الشبكة، ففي هذه الحالة يمكن أن يحدث هجوم دون الشعور به.
3. النظم المدمجة بين الاثنين(Hybrid IDS): يعتبر الدمج بين الاثنين أكثر فعالية إذ يتكاملان في مهمتهما في اكتشاف الهجوم ودقة تحديده.

وفي كل الحالات وبغض النظر عن نظام اكتشاف الاختراقات المستعمل، فلا بد أن يكون هناك فريق مختص يتابع الإنذارات الصادرة من النظام، ويقوم بالإجراءات اللازمة للحد أو لصد الهجوم وإيقافه.

2. 1. 3 دراسة حالة: نظام SNORT

نتعرض هنا لدراسة نظام SNORT لاكتشاف الاختراقات الذي يعتبر اليوم من أشهر نظم اكتشاف الاختراقات. طور SNORT في نهاية التسعينات ويدمج بين استعمال السياسات الأمنية والتقنية، ففي السياسة الأمنية نحدد المسموح به والمحظور من حركة الطرود

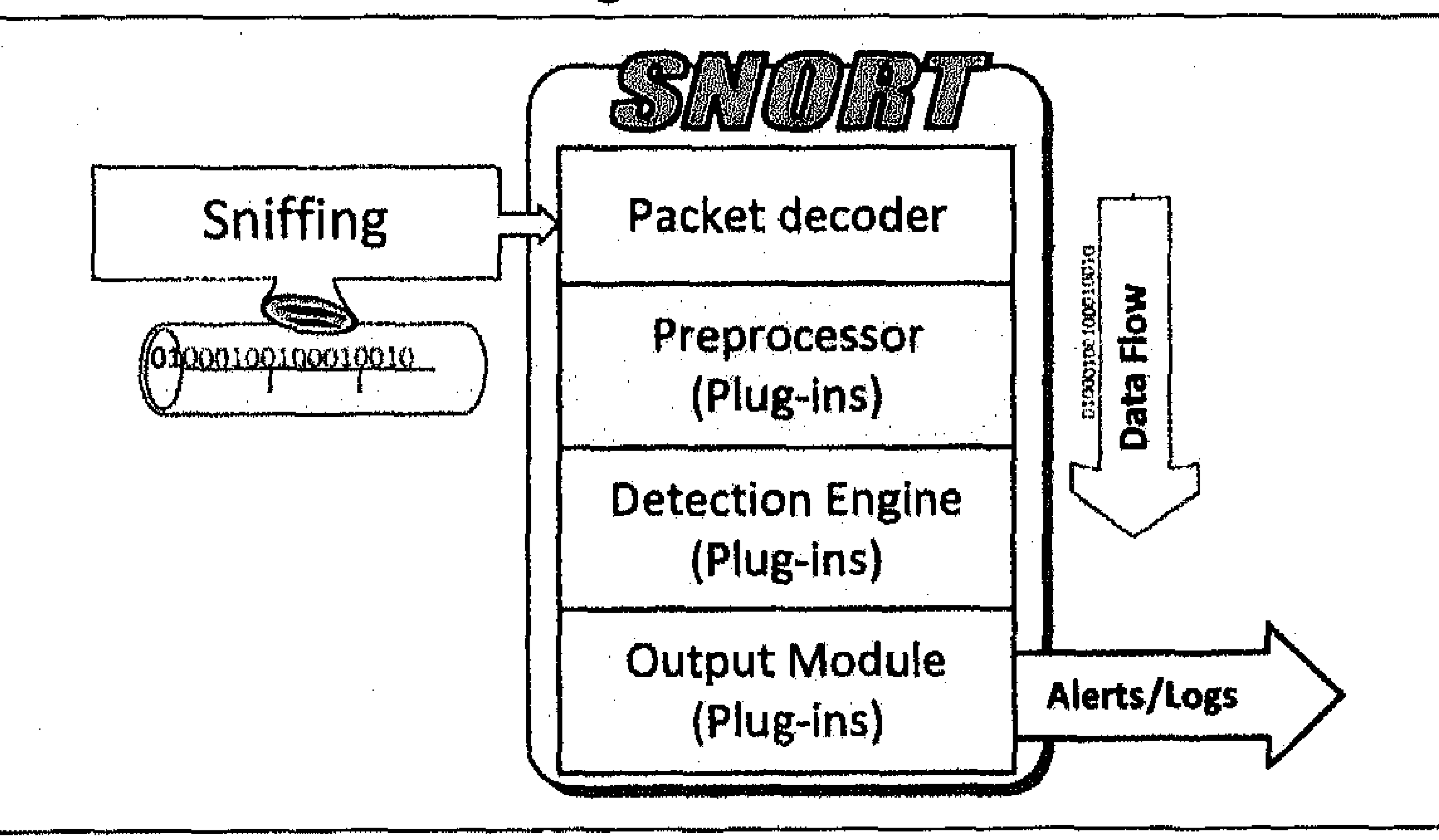
الداخلة والخارجة، وأيضًا ما الذي يعتبر من الأنماط هجومًا في هذه الطرود. تكون هذه السياسات في شكل قواعد يستعملها النظام في اكتشاف الهجومات، ويقع تحديثها للأخذ بعين الاعتبار الثغرات الجديدة المكتشفة. يقوم النظام بتوليد إنذار أمني كلما وقع اكتشاف هجوم لمدير الشبكة الذي يقوم بالإجراءات اللازمة للحد أو لصد الهجوم. يتميز SNORT بصغر حجم برنامجه وتوافقيته مع أغلب الأنظمة وبسرعته، وسهولة ضبطه، وبكونه مفتوح المصدر ومجانيًا. ويمكن أن يعمل SNORT كبرنامج تنصت على الشبكة أو برنامج تحليل لآثار هجوم معين، أو كنظام اكتشاف اختراقات بوضعيه النشط في الوقت الحقيقي أو الخامل.

3. 1. 3 عمارة SNORT

يتكون SNORT من أربع أجزاء أساسية وهي (انظر الشكل الموالي)

1. مفك الطرود الذي يتعرف على نوعية البرتوكول وحقوله.
2. المعالج الأولي الذي يعالج الطرود ويهيؤها للدخول إلى محرك الاكتشاف.
3. محرك الاكتشاف الذي يبحث عن أنماط الهجوم في الطرود باستعمال قواعد SNORT.
4. مصدر الإنذارات الذي يكتب الإنذارات في السجل أو يرسلها لمدير الشبكة.

شكل 11.4 عمارة SNORT



4. 1. 3 قواعد SNORT

يمتلك SNORT قواعد مرنة جدًا، وسهلة التغيير على عكس كثير من نظم اكتشاف الاختراقات التجارية. فيما يلي مثال على قاعد اكتشاف حصان طروادة SUBSEVEN.

شكل 4. 12قاعدة SNORT.

```
alert tcp $EXTERNAL_NET 27374 -> $HOME_NET any (msg:"BACKDOOR
  subseven 22"; flags: A+; content: "|0d0a5b52504c5d3030320d0a|";
  reference:arachnids,485;
  reference:url,www.hackfix.org/subseven/; sid:103;
  classtype:misc-activity; rev:4;)
```

فالعناصر قبل الأقواس المفتوحة تكون رأس القاعدة، والعناصر التي ما بين القوسين هي خيارات القاعدة. ولنقم بتفصيل هذه القاعدة كتطبيق عملي لفهمها.

رأس القاعدة:

- alert : العمل الذي يجب القيام به فهنا تقوم هذه القاعدة بالإنذار، وثمة خيارات أخرى السماح بالمرور، أو الكتابة في السجل وغيرها.
- tcp : البروتوكول المعني بالأمر وهو هنا TCP ويدعم SNORT أهم البرتكولات الأخرى المستعملة في الشبكة كـ UDP وICMP وIP.
- $EXTERNAL_NET : متغير يحمل عنوان مصدر أو مرسل الطرود، ويمكن تحديده كعنوان IP ثابت.
- 27374 : منفذ المصدر وهو هنا محدد بـ27374 ويمكن جعله مطلقًا باستعمال كلمة any وأيضًا يمكن استثناء منفذ معين باستعمال عامل! فمثلا (21!) معناه السماح لكل منفذ عدا منفذ 21 تحديد مجال معين للمنافذ مثلا (1:1024) من المنفذ 1 إلى المنفذ 1024
- -> : اتجاه البيانات، وهو هنا من الخارج للداخل، ويمكن استعمال <> لتحديد أن القاعدة تعمل في الاتجاهين على البيانات الداخلة والخارجة من الشبكة.
- $HOME_NET : متغير يحمل عنوان المستقبل للبيانات، ويمكن تحديده بقيمة IP ثابتة.
- any : منفذ المستقبل وهنا قيمته مطلقة أي على أي منفذ استقبال.

قائمة الخيارات:

- "msg:"BACKDOOR subseven 22;: الرسالة التي ستظهر في ملف السجل.
- flags: A+;: أعلام TCP هناك جملة من الأعلام مثل SA,SA+,!R,SF* وغيرها.
- content: "|0d0...0a|";: نمط البيانات التي سيتحقق من وجودها أو عدمه في الطرود.
- reference...;: إشارة مرجعية لتفصيل هذه القاعدة.
- sid:103;: معرف القاعدة.
- classtype: misc-activity;: نوع القاعدة، هناك عدة أنواع غيرها.
- rev:4;: رقم مراجعة القاعدة.

وهناك عدة خيارات أخرى مفصلة في مراجع SNORT على موقع sourcefire.com.

يعرض الشكل الموالي قواعد عملية لاكتشاف الهجومات الفعلية. فالقاعدة الأولى تكتشف عملية هجوم على خادم MS-SQL، والقاعدة الثانية خاصة بخادم الويب والثالثة خاصة بخادم نقل الملفات.

شكل 13.4 أمثلة عملية لصد هجومات حقيقية.

```
- alert tcp $EXTERNAL_NET any -> $SQL_SERVERS 1433
  (msg:"MS-SQL xp_cmdshell - program execution"; content:
  "x|00|p|00|_|00|c|00|m|00|d|00|s|00|h|00|e|00|l|00|l|00
  |"; nocase; flags:A+; classtype:attempted-user;
  sid:687; rev:3;) caught compromise of Microsoft SQL Server
- alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS 80
  (msg:"WEB-IIS cmd.exe access"; flags: A+;
  content:"cmd.exe"; nocase; classtype:web-application-
  attack; sid:1002; rev:2;) caught Code Red infection
- alert tcp $EXTERNAL_NET any -> $HOME_NET 21 (msg:"INFO
  FTP \"MKD / \" possible warez site"; flags: A+;
  content:"MKD / "; nocase; depth: 6; classtype:misc-
  activity; sid:554; rev:3;) caught anonymous ftp server
```

كما يقسم SNORT هذه القواعد إلى عدة ملفات بحسب الخدمة، مثل ftp,telnet,http وغيرها من الخدمات الأخرى، وذلك لتحسين الفعالية وتسريع عملية اكتشاف الهجومات.

4 - نظم اكتشاف البرامج الخبيثة

الفيروسات الحاسوبية هي برامج أو أوامر ذات حجم صغير لها تأثيرات ضارة على برامج أخرى، أو أنظمة المعلومات وشبكاته بشكل عام، وذلك عن طريق إحداث تعديل فيها، وهذا التعديل قد يكون بنسخ نفسه داخلها، ومن ثم يتم تنفيذ أوامره الضارة أثناء تشغيل البرنامج، كما أن له القدرة على الانتشار من أي وسط تخزين أو جهاز إلى آخر. ويمكن تقسيمها إلى صنفين، الصنف الأول: يتطلب أن تستضاف من قبل البرنامج المراد تدميره، والصنف الآخر : هو برامج المستقلة بذاتها.

تحتوي أغلب الفيروسات على ثلاثة مكونات رئيسة، هي:

1– طريقة التزايد (حتى تستطيع أن تصل إلى برامج وأجهزة أخرى).

2– المؤقت الذي يؤدي لحدوث أمر ما في موعد محدد.

3– السلوك المربك الذي يحدث عند تشغيل المؤقت (وهو عمل الفيروس).

ويمر الفيروس خلال فترة حياته بالأطوار التالية:

- **طور السكون**: Dormant phase: ويكون فيها الفيروس غير نشط، وعلى استعداد للعمل عند حدوث أي حدث، مثل الوصول الجهاز إلى وقت أو تاريخ محدد، أو فتح ملف، أو تشغيل برنامج، أو ملء الذاكرة بحجم معين من البيانات.
- **طور الانتشار Propagation phase**: في هذه المرحلة يقوم الفيروس بنسخ نفسه في صورته غير النشطة داخل برامج أخرى، أو داخل أماكن أخرى من ذاكرة النظام. وفي هذه الحالة سوف تحتوي البرامج المصابة على نسخة طبق الأصل للفيروس، والذي بدورة سوف يمر بطور انتشار جديد.
- **طور الإثارة وقدح الزناد Triggering phase**: في هذا الطور سوف يكون الفيروس في حالة نشاط لأداء الوظيفة المحددة له، وذلك عند حدوث حدث معين في النظام المصاب بالفيروس.

- **طور التشغيلExecution phase** : هنا يؤدي الوظيفة المحددة، والتي قد تكون ضارة، مثل: تعطيل البرامج عن أداء وظائفها على الشكل المطلوب، وإتلاف البيانات أو غيره، أو قد تكون غير ضارة، مثل: إرسال رسائل على شاشة الجهاز.

4.1 بناء الفيروس

يمكن أن يقحم الفيروس نفسه داخل البرنامج القابل للتشغيل، إما في بدايتها أو في نهايتها، بحيث عندما يتم استدعاء البرنامج للتشغيل فإن أوامر الفيروس تنفذ أولاً، ومن ثم يتم تنفيذ أوامر البرنامج المصاب. وكما هو مبين في التركيب العام للفيروس الموضح في الشكل رقم 1 (كوهي 94 COHE)، فإن أمر الفيروس V مضاف إلى بداية البرامج المصابة، حيث إن من المفترض أن يكون المدخل للبرنامج هو السطر الأول من أوامره عندما يتم تشغيله، وبهذا يبدأ البرنامج المصاب بتنفيذ أوامر الفيروس كما يلي: السطر الأول من الأوامر يتضمن الانتقال إلى البرنامج الأساس للفيروس، أما السطر الثاني فهو عبارة عن علامة تستخدم بواسطة الفيروس، لتحديد ما إذا تمت إصابة البرنامج بهذا الفيروس أم لا، وعندما يستدعى البرنامج للتشغيل يتحول التحكم مباشرة إلى البرنامج الأساس للفيروس، عندها يقوم الفيروس أولاً بالبحث عن الملفات القابلة للتشغيل، والتي لم تصب فيصيبها بنفسه، ومن ثم قد يقوم بتنفيذ بعض المهام المحدد على النظام والتي يتم تنفيذها في كل مرة يتم تشغيل هذا البرنامج المصاب، أو قد تكون قنبلة موقوتة تنفذ تحت شرط معين. وفي النهاية يتم تحويل التحكم إلى البرنامج الأصلي المصاب، وإذا كانت فترة تشغيل الفيروس أو فترة تنفيذ وضيفته قصيرة جدًا فإن المستخدم لن يلاحظ وجود الفيروس.

شكل 14.4 فيروس في شكله المبسط.

```
program V :=

{goto main;
    1234567;
    subroutine infect-executable :=
              {loop:
                  file := get-random-executable-file;
                  If(first-line-of-file = 1234567) then
                     goto loop;
                  else  prepend V to file; }

    subroutine do-damage :=
               { whatever damage is to be done }

    subroutine trigger-pulled :=
                {return true if some condition holds}

main:       main-program :=
                { infect-executable;
                if trigger-pulled then do-damage;
                goto next;}
next:
}
```

باتباع الفيروس لهذه الطريقة في إقحام نفسه وإصابة البرامج فإنه من السهل اكتشافه، وذلك باعتبار أن طول البرنامج بعد إصابته بالفيروس أطول منه قبل الإصابة، وبالمقارنة بين الحالتين يتبين وجود أوامر مضافة للبرنامج الأصلي والتي هي عبارة عن الفيروس نفسه. ولكي يتغلب الفيروس على هذه الطريقة في اكتشافه، فإنه يقوم باستخدام تقنية ضغط البرامج بحيث يكون طول البرنامج المصاب مساو لطول البرنامج الأصلي. يبين الشكل رقم 2 (كوهي COHE 94) العملية المنطقية وراء هذه الأسلوب موضحًا أسطر الأوامر المفتاحية مرقمة. ويمكن شرح هذه العملية كما هو مبين في الشكل رقم 3 (كوهي COHE 94) وذلك بافتراض أن P_1 يمثل البرنامج المصاب بالفيروس VC. وعندما يستدعى هذا البرنامج للتشغيل، يتحول التحكم إلى الفيروس، والذي يقوم بدوره بإجراء العمليات التالية:

1. لكل ملف غير مصاب يتم العثور عليه، يقوم الفيروس أولاً بضغط هذا الملف ليصبح P'_2، بحيث يكون أقل من النسخة الأصلية بمقدار طول الفيروس.
2. يتم إقحام الفيروس نفسه في مقدمة البرنامج المضغوط.
3. يتم فك ضغط النسخة المضغوطة من البرنامج الأصلي المصاب P'_1

شكل 15.4 العملية المنطقية للفيروس معتمد على ضغط البرنامج المصاب.

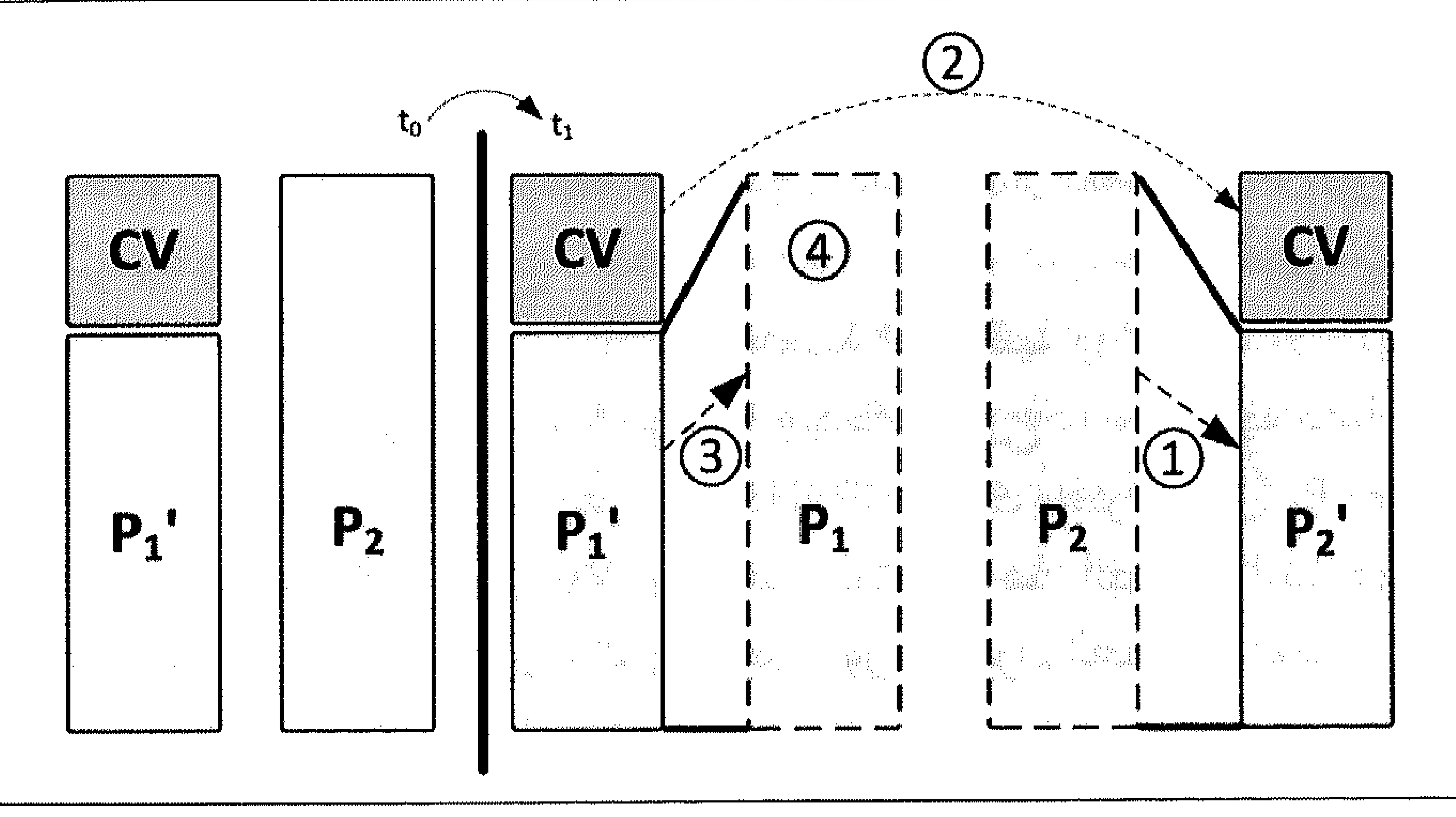
```
program CV :=

{goto main;
    01234567;

    subroutine infect-executable :=
        {loop:
         file := get-random-executable-file;
         If(first-line-of-file = 1234567) then goto loop;
          (1)  compress file;
          (2)  prepend CV to file;
        }

Main;   main-program :=
        {if ask-permission then infect-executable;
          (3)  uncompress rest-of-file;
          (4)  run uncompressed file;}
        }
```

شكل 16.4 فيروس ضغط.



4.2 أنواع البرامج الضارة وذلك على حسب تكوينها ووظائفها

1- حصان طروادةTrojan Horse

هي أوامر تختفي داخل برامج مفيدة، وهذه الأوامرتقوم بعمل سيئ غير مشروع. حصان طروادة يتكون عندما يتم إدخال تلك الأوامر أثناء كتابة البرنامج بعكس الفيروسات التي هي عبارة عن أوامر تدرج في برامج موجودة في الأجهزة.
كما أن أحصنة طروادة لا تنتظر وجود عامل (كوقت محدد) فبمجرد أن يقوم بتشغيل البرنامج الذي يحتوي عليه يبد أمباشرة في العمل كما أن حصان طروادة عادة ما يستخدم كما لو أنه جاسوس وذلك بإرساله لعدد من البيانات المهمة المطلوبة مثل كلمات المرور.

2- **الفيروسات Virus** : هي مجموعة من الأوامر عند تنفيذها تقوم بنسخ نفسها داخل أحدالبرامج المعروفة الموجودة في النظام وذلك منخلال التصاقها به تنتهي بـCOM أو SYS أو EXE مثل MSWord، أوأحد المكونات الكبرى في نظام التشغيل، وهو القطاع الجذري بحيث أنه عند تشغيل البرنامج سوف يبدأ الفيروس في العمل.

3- **البكتيريا Bacterium** : هي برامج صغيرة مستقلة بذاتها لا تلتصق ببرامج أخرى تقوم بتكرار نفسها مسببة استهلاكًا لقدرات ووقت بعض مصادر النظام.

4- **الدودة Worms**: هي برامج صغيرة جداً تقوم بتكرار نفسها وذلك بإنشاء نسخ لها على أجهزة أخرى عبر الشبكة، وهي مشابهة للبكتريا ،لكنها تكرر نفسها فوق الشبكة، لكن البكتريا تظل في جهاز واحد، وبذلك هي لاتحتاج إلى برنامج مضيف.

5- **الباب الخلفي Trapdoor**: نقطة عبور البيانات من وإلى النظام غير مشروعة، وغير مدونة تكتب داخل برنامج ،وعادة ما يكون لغرض فحص العيوب Debugging وهذه النقطة تستغل من المهاجمين، أوالمعتدين كمنفذ غير آمن.

6- **القنبلة المنطقيةLogic bomb**: هي عبارة عن أوامر مشبوهة بداء بالعمل عند حدوث حدث ما Trigger في النظام أو عند توقيت معين.

4 .3 أنواع الفيروسات من حيث تركيبتها

هناك سجال كبير بين مصممي الفيروسات الحاسوبية، وبين شركات تصنيع برمجيات مكافحة الفيروسات منذ ظهور أول فيروس حاسوبي. فكلما ظهر فيروس وتم تطوير مضاد له خرج نوع من الفيروسات جديد يتغلب على مضادات الفيروسات، وهكذا دواليك. في عام 1993 قام ستفنسون (ستفينسون STEP 93) باقتراح التصنيف التالي للفيروسات الحاسوبية:

- الفيروسات الطفيلية Parasitic : وهي الفيروسات المعتادة والمستخدمة بكثرة، وهي تقوم بإضافة نفسها في الملفات القابلة للتنفيذ، إضافة إلى قدرتها على التكاثر.
- الفيروسات المقيمة في ذاكرة الجهاز: وهي فيروسات تقيم في الذاكرة الرئيسة كجزء من البرامج المقيمة.
- فيروسات قطاع التشغيل (Boot Sector): وهي تصيب قطاع تحميل وتشغيل النظام بحيث تنتشر حال تحميل النظام من القرص الموجود فيه الفيروس.
- الفيروس الخفي: وهو نوع من الفيروسات له القدرة على التخفي من برامج مضادات الفيروسات.

4 .4 كيفية تكوين الفيروس الحاسوبي وآلية عمله بشكل مختصر

- استبدال أي تعليمة instruction في البرنامج على الموقع Location x بأمر (Jump to y) بحيث يكون y هو عنوان مكان خالٍ(Free place location)ثم،
- كتابة أوامر الفيروس في مكان يبدأ بالعنوان y ثم:
- القيام بوضع الأمر instruction الذي كان أصلاً موجوداً على العنوان x في آخر أوامر الفيروس متبوعاً بأمر ((jump to (x+1) بحيث لا يحس المستخدم، وبذلك عند تنفيذ الأمر الذي كان على العنوان x يقوم الفيروس بالعمل.

4.5 طرق فحص الفيروسات والقضاء عليها Virus checker

في الغالب لا أحد يقوم بفحص البرامج حتى وإن أراد أحد عمل ذلك، فإنه قد يكون من المستحيل الإخبار بما سيعمله كالأمر في البرنامج بمجرد النظر إلى تركيبته program'scodes، وبالتالي لا يمكن الإخبار بأن برنامجًا ما له تأثير جانبي سيء بمجرد فحص الـcodes، هذا على افتراض السماح بالاطلاع علىprogram codes من قبل مصممه. فالطريقة المثلى في مكافحة الفيروسات هي الوقاية منها، بحيث تمنع وصول الفيروسات إلى نظام المعلومات،لكن في الواقع يبدو أن هذه الطريقة سوف تكون صعبة التحقيق لفترة زمنية طويلة، لذا فإن الطريقة الجيدة الأخرى تتمثل في القدرة على عمل التالي:

- اكتشاف الفيروس: فبمجرد تعرض النظام للفيروس فينبغي المسارعة بمعرفة الضرر وتحديد الفيروس.
- تحديد هوية الفيروس: فبمجرد اكتشاف الضرر الذي أحدثه الفيروس فينبغي تحديد هوية الفيروس الذي أصاب البرنامج
- حذف الفيروس: عند تحديد هوية الفيروس يتم القيام بحذف جميع آثاره من الملفات المصابة به.

وتقوم برامج فحص الفيروسات(Virus checkers programs) لتحقيق ذلك بالتالي:

- الاعتماد على معرفة تسلسل الأوامر (instruction sequence) لكثير من أنواع الفيروسات، وبذلك بفحص جميع الملفات على الاسطوانة والأوامر في الذاكرة لكشف هذه الصيغة من تسلسل الأوامر، ومن ثم إظهار تنبيه في حال وجود مثل هذه الصيغة، لذلك لابد من عمل تحديث (update) لهذه البرامج الكاشفة للفيروسات بشكل دوري، وذلك بإضافة صيغ جديدة لتسلسل أوامر فيروسات جديدة إلى هذه البرامج. لذلك فإن أحد مكونات برامج مكافحة الفيروسات هو الماسح (Scanner).
- لكن صانعي الفيروسات لاحظوا هذه الطريقة لكشف الفيروسات فقاموا بتصميم نوع جديد من الفيروسات يسمى بوليمورفيك (polymorphic Virus)، وهذا النوع يقوم بتغيير ترتيب أوامره (its

instructions) أو تغيرها إلى دوال أخرى لها نفس الوظيفة، وذلك في كل مرة يقوم بنسخ نفسه.

- هناك برامج فحص فيروسات أخرى تنتهج آلية مختلفة في كشف الفيروسات بحيث تقوم بتسجيل محتويات الذاكرة على فقرات take a snapshot of disk storage والخاص بالملفات، مثل طول الملف. ومن ثم إظهار رسالة تحذير عن حديث تغيير لهذه المعلومات، لكن تقوم بعض الفيروسات بضغط البرنامج المحتواة فيه بحيث يحافظ على طوله الأصلي.

اكتشف المتخصصون في البرامج المضادة للفيروسات أن بعض الفيروسات مثل فيروس Melissa يحتوي على أرقام تعريف يمكن تتبعها هيGlobal Unique Identifier أو (GUID) وهذه تعتبر بمثابة بصمة لكل برنامج، فإذا كان لكل مستند (ملف) ID خاصة به فمن السهل أن نتعرف على مؤلف هذا الملف مما يساعد في التعرف على صانع الفيروس.

5 - أمن التطبيقات

5.1 أمن البريد الإلكتروني

تعتبر خدمة البريد الإلكتروني من أقدم وأشهر وأوسع خدمات الإنترنت انتشارًا. ومع هذا فإن هذه خدمة بصيغتها الأصلية ليست آمنة. خدمة البريد الإلكتروني هي في واقع الأمر خدمة تبادل البطاقات البريدية لا الرسائل البريدية. فكل واحد يمكنه الاطلاع على محتوى البطاقة إن أراد ذلك، إما عند التبادل على الشبكة، أو عند خوادم بريدي المرسل والمستقبل. والمطلوب هو جعل هذه الخدمة أقرب ما تكون لخدمة البريد الآمن الحقيقية. هناك جملة من المتطلبات الأمنية التي من وجودها وهي السرية والتأكد من هوية المرسل، وسلامة الرسالة، وضمان عدم إنكار المرسل إرساله الرسالة. ظهرت جملة من النظم تسعى لتوفير هذه المتطلبات، أهمها: نظام الخصوصية الفائقة PGP الذي سبق أن تعرضنا لبعض خصائصه في هذا الفصل ونظام S/MIME وهي النسخة الآمنة لنظام البريد الإلكتروني متعدد الأغراض على الانترنت Multipurpose Internet Mail Extensions (MIME)).

1. 1. 5 نظام PGP

نظام PGP هو الأكثر استعمالاً على نطاق الإنترنت. طور النظام فيل زمرمان الذي اختار أشهر أنظمة التشفير ودمجها في برنامج واحد، يدعم PGP نظم يونكس ووندوز وماكنتوش وأميغا، وهو في البداية كان مجانيّاً أما الآن فيوجد له نسخ تجارية. يقدم النظام خدمة السرية، والتأكد من هوية المرسل، وسلامة الرسالة، كما يدعم تقنيات الضغط والتوافقية.

- خدمة التأكد من الهوية والسلامة:

1. يحضر المرسل رسالته.
2. يستعمل النظام خوارزمية توليد البصمة SHA-1 لتوليد بصمة الرسالة بطول 160 بت.
3. يتم تشفير هذه البصمة بخوارزمية RSA باستعمال المفتاح الخاص للمرسل والناتج يرفق مع الرسالة.
4. يستعمل المستقبل RSA أو DSS والمفتاح العام للمرسل لفك شفرة البصمة وبالتالي استخراج البصمة.
5. يحسب المستقبل بصمة الرسالة التي وصلته، ويقارنها مع البصمة المستخرجة، فإن توافقت فنكون قد تأكدنا من سلامة الرسالة، ومن هوية المرسل وإن لا فلا.

- خدمة سرية الرسالة:

1. يولد المرسل الرسالة وعددًا عشوائيًّا بطول 128 بت ليستعمل في تشفير هذه الرسالة فقط.
2. يتم تشفير الرسالة باستعمال هذا المفتاح، وأحد خوارزميات التشفير التماثلي التالية CAST-128 أو IDEA أوDES3
3. يتم تشفير مفتاح تشفير الرسالة المستعمل بخوارزمية RSA باستعمال المفتاح العام للمستقبل، ثم يتم إرفاقه بالرسالة.
4. يستعمل المستقبل مفتاحه الخاص لفك الشفرة واستخراج مفتاح تشفير الرسالة.
5. ثم يستعمل هذا المفتاح المستخرج لفك شفرة الرسالة.

يمكن استعمال خدمتي السلامة، والتأكد من الهوية، وخدمة السرية مع لنفس الرسالة وذلك بـ:

1. صناعة توقيع رقمي وإرفاقه بالرسالة.
2. تشفير التوقيع الالكتروني والرسالة معًا بمفتاح تشفير معين.
3. تشفير مفتاح التشفير بـRSA وإرفاقه بالرسالة.

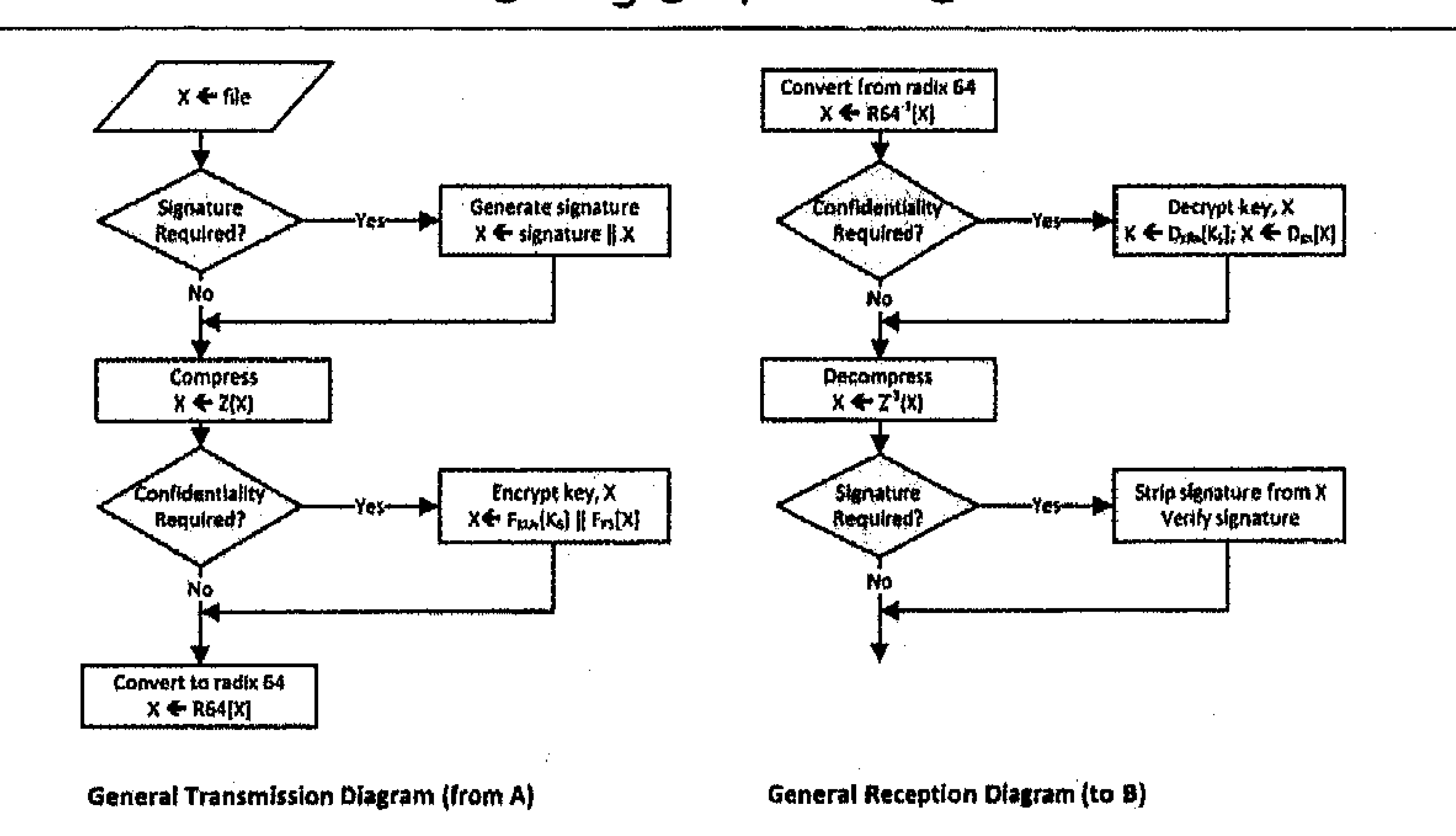
- خدمة ضغط البيانات والتوافقية:

يضغط PGP الرسالة بعد إمضائها، ولكن قبل تشفيرها ويستعمل خوارزمية ZIP في ذلك. كما أنه عندما نستعمل PGP فإن الناتج سيكون بيانات على الترميز الثنائي Binary data ، والبريد إنما يتعامل مع النصوص فقط Ascii data ولهذا فإن PGP يدعم التوافقية بين النظامين، ويستعمل لهذا الغرض خوارزمية RADIX-64 التي تحول كل ثلاث بايت إلى 4 حروف كما تضيف كود التحقق CRC. كما أن PGP يقوم بتقسيم الرسالة إلى كتل نصية متعددة فيما لو كان حجم الرسالة كبيرًا جدًا.

يلخص الشكل الموالي العمليات السالفة الذكر عند المرسل A (انظر الجزء الأيسر من الشكل 4. 17) والمستقبل B (انظر الجزء الأيمن من الشكل 4.17).

أما بالنسبة لإدارة المفاتيح العامة فسبق التفصيل في ذلك في الفقرة 5.3.1 من الفصل الثالث.

شكل 4. 17 تبادل رسائل PGP.



2. 1. 5 نظام S/MIME

كان البريد الإلكتروني عند أول ظهوره لا يدعم إلا النصوص كما هو مبين في وثيقة الإنترنت RFC822، ثم وقع تطويره لدعم أكثر من محتوى كالصور، والوسائط المتعددة، ليعرف النظام باسم MIME. ثم وقع إضافة الخدمات الأمنية في نسخة MIME الآمنة S/MIME. يعمل هذا النظام في عدة برامج البريد الإلكتروني الحديثة كبرنامج Outlook وبرنامج Netscape وغيرها، يوفر S/MIME عدة خدمات أهمها:

1. خدمة السرية: بتشفير الرسالة بمفاتيح معينة.
2. خدمة التوقيع الالكتروني: بتوليد البصمة وتوقيعها رقميًا.
3. خدمة سلامة الرسالة: بتوليد بصمة الرسالة وإرسالها مع الرسالة غير المشفرة للمستقبل.
4. دمج خدمة السرية مع التوقيع الالكتروني: بتوقيع الرسالة، ثم تشفير التوقيع والرسالة معًا.

يستعمل S/MIME عدة خوارزميات للتشفير، وتوليد البصمة، والتوقيع الالكتروني، ويملك آلية اختيار بين هذه الخوارزميات. فبالنسبة لتوليد البصمة فإنه يستعمل خوارزميتي SHA-1 وND5. وبالنسبة للتوقيع الإلكتروني فإنه يستعمل خوارزميتي DSS وRSA. ولتشفير مفتاح تشفير الرسالة يستعمل خوارزميتي RSA والجمل ElGamal. وأخيرًا لتشفير الرسالة يستعمل DES3 وRC2/40 وغيرها من خوارزميات التشفير التماثلي. أما في إدارة المفاتيح فإنه يستعمل نظامًا هَجِينًا يستعمل معيار X.509 ونظام PGP.

2. 5 أمن الشبكة العنكبوتية

تستخدم الشبكة العنكبوتية اليوم في معظم الأنشطة الإنسانية. وتحولت من شبكة تبادل معلومات مفتوحة إلى شبكة تبادل خدمات مهمة كالتجارة الإلكترونية، والحكومة الإلكترونية، والتعليم عن بعد، وكسائر الخدمات الأخرى فإن خدمة الويب مهددة بجملة من الهجومات التي تمس السرية، والسلامة، والتأكد من الهوية، والتوفيرية. ظهرت عدة

حلول وآليات أمنية لخدمة الويب أهمها آلية طبقة المكبس الآمنة SSL/TLS وبرتوكول العمليات التجارية الآمنة SET.

1. 2. 5 طبقة المكبس الآمنة SSL/TLS

طورت شركة نتسكيب Netscape طبقة المكبس الآمنة SSL(Socket Secure Layer) لتصبح بعد ذلك معيارًا في شبكة الإنترنت يعرف باسم أمن طبقة النقل TLS (Transport Layer Security) حيث يستعمل برتوكول TCP لنقل البيانات.

عمارة SSL:

تتكون SSL من طبقتين اثنين من البروتوكولات، وهما:

- جلسة SSL:

وهي عبارة عن ربط بين العميل والخادم عن طريق بروتوكول المصافحة HandShake ، حيث يتم الاتفاق على جملة من عوامل التشفير، يمكن أن يتشارك فيها أكثر من اتصال SSL.

- اتصال SSL:

وهي عبارة عن اتصال عابر غير دائم من نوع الند للند، ويرتبط مع جلسة SSL واحدة.

وتتكون عمارة SSL من العناصر التالية (انظر إلى الشكل 4. 18)

- بروتوكول السجلات SSL Record Protocol:

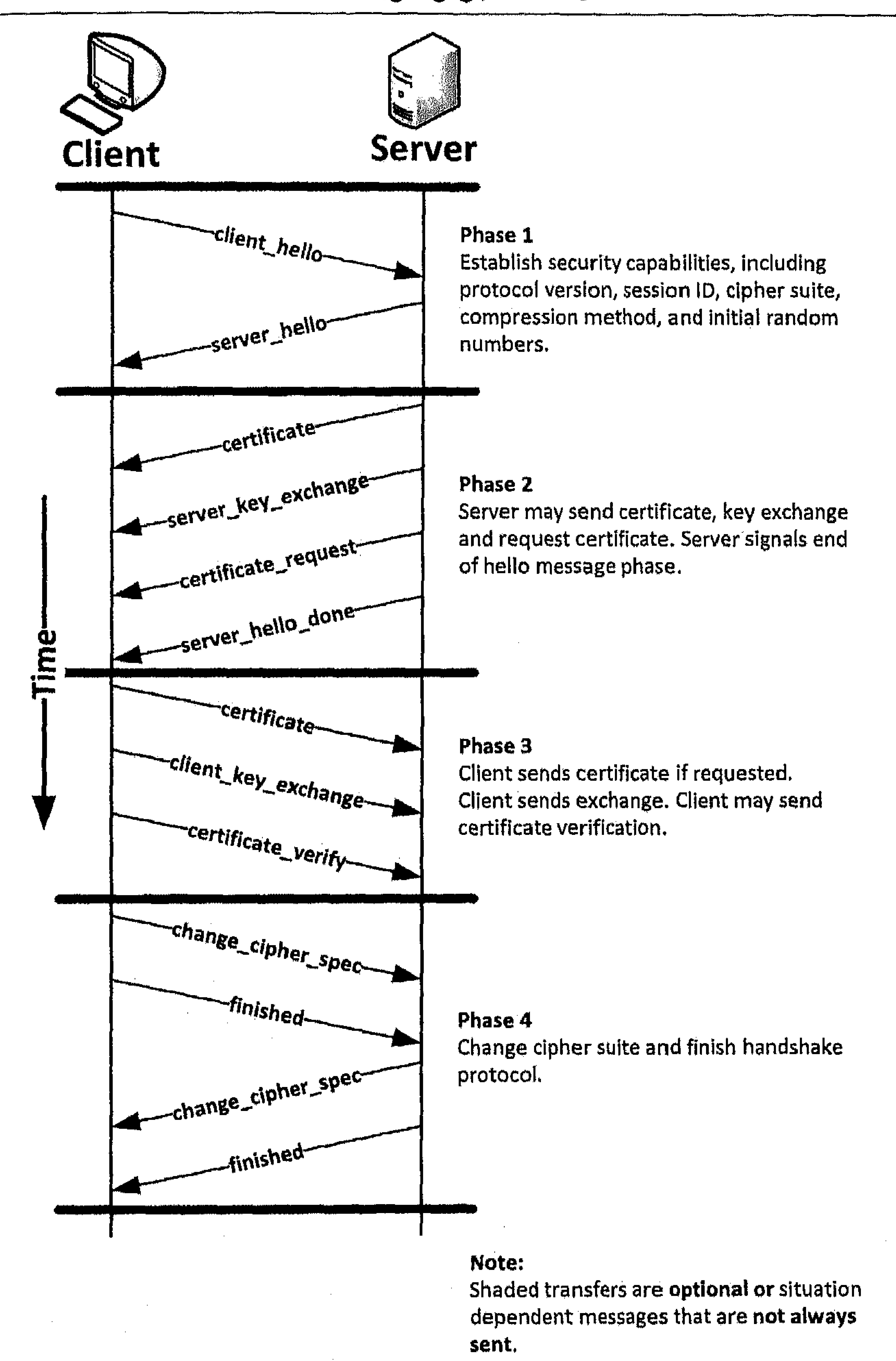
يوفر خدمتي سرية وسلامة البيانات؛ إذ يستعمل التشفير التماثلي بمفتاح سري مشترك يعرف عند الاتصال من خلال برتوكول المصافحة. يستعمل عدة خوارزميات مثل IDEA, RC2-40, DES-40, DES, 3DES, Fortezza, RC4-40, RC4-128، كما أنه يتم ضغط الرسالة قبل التشفير. أما في خدمة السلامة فإنه يستعمل كود MAC باستعمال المفتاح السري المشترك، وهي شبيهة بـHMAC لكن بنظام حشو بيانات مختلف Data Padding.

شكل 19.4 عمارة SSL

SSL Handshake Protocol	SSL Change Cipher Spec Protocol	SSL Alert Protocol	HTTP
SSL Record Protocol			
TCP			
IP			

- برتوكول تغيير التشفير: SSL Change Cipher specific protocol
واحد من ضمن ثلاثة البرتوكولات التي تستعمل برتوكول السجلات، ويقوم بتغيير وتحديث نظام التشفير في برتوكول السجلات
- برتوكول الإشعار Alert Protocol:
يقوم هذا البرتوكول بإشعار جهتي الاتصال بالإنذارات المتعلقة بـSSL، وهي إما أن تكون إنذار تحذير وتنبيه، أو إنذارًا بخطأ فادح. وتعم عدة أنواع من الأخطاء المتعلقة بعملية التشفير، وتوليد البصمة، والضغط، وفك الضغط، وإدارة الشهادات Certificates كأن تكون الشهادة غير معتمدة، أو ملغاة أو غير ذلك من الأخطاء. وككل بيانات SSL فهذه الإنذارات تكون مضغوطة ومشفرة.
- برتوكول المصافحة Handshake Protocol:
يمكن هذا البرتوكول الخادم والعميل من التأكد المتبادل للهوية، والاتفاق على خوارزميات التشفير، وتوليد البصمة، وأيضًا مفاتيح التشفير التي ستستعمل في الاتصال. يتم هذا الاتفاق بتبادل سلسلة من الرسائل عبر أربع مراحل وهي: إنشاء القدرة الأمنية للجهتين، ثم تتم عملية التأكد من هوية الخادم، وتبادل المفاتيح ثم عملية التأكد من العميل، وتبادل المفاتيح ثم عملية إنهاء تبادل الرسائل (انظر إلى الشكل 20.4).

شكل 20.4 بروتوكول المصافحة.



تم اعتماد SSL في نسختها الثالثة كمعيار للإنترنت في وثيقة RFC2246. وهي عبارة عن SSLv3 مع بعض الفروقات الطفيفة، مثل: إضافة رقم النسخة واستعمال HMAC وإضافة أنواع أخرى من الإنذارات، وبعض التغييرات في المفاوضة على الشهادات، وأيضًا طرق حشو البيانات.

بروتوكول العمليات التجارية الآمنة SET

يهدف بروتوكول SET إلى إجراء عمليات الدفع باستعمال البطاقات الائتمانية على الإنترنت بشكل آمن. طورت البروتوكول شركة ماستر كارد وشركة فيزا وغيرها من الشركات الأخرى، وهو عبارة عن مجموعة من بروتوكولات الأمن التي توفر الخدمات التالية:

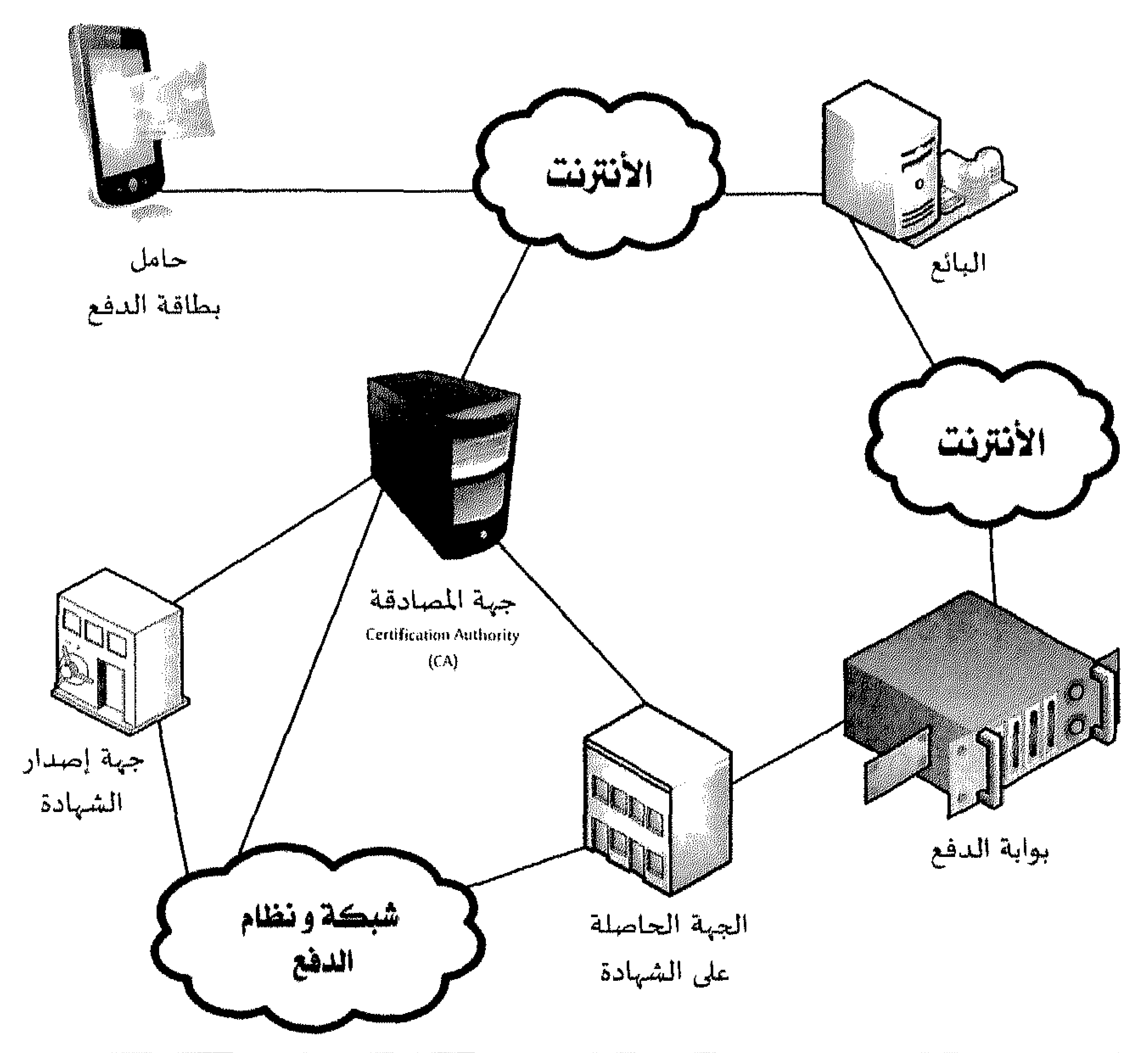
1. اتصالات آمنة بين مختلف الجهات التجارية.
2. الثقة بين أطراف الاتصال باستعمال شهادات معيار X.509v3.
3. الخصوصية، وذلك بقصر معرفة المعلومات على الأطراف المعنية بها فقط.

- عمارة بروتوكول SET:

تتكون عمارة SET من عدة أطراف هي (انظر إلى الشكل الموالي):

1. البائع على الإنترنت Merchant
2. جهة المصادقة على الشهادات CA
3. جهة إصدار الشهادة Issuer
4. الجهة الحاصلة على الشهادةAcquirer
5. شبكة و نظام الدفع Payment Network
6. صاحب بطاقة الدفع Cardholder

شكل 4. 21عمارة SET



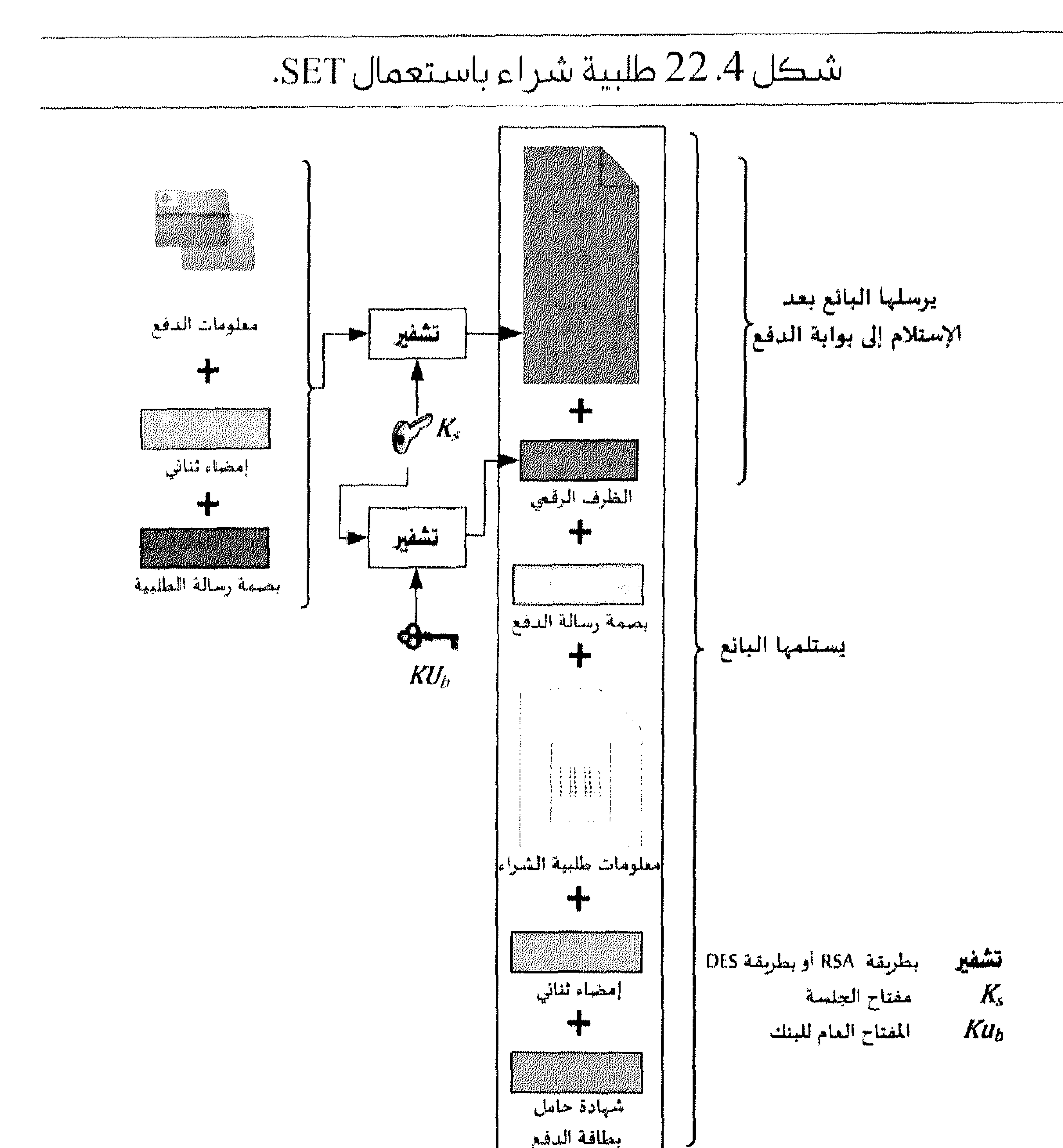
يقوم العميل بتوليد رسالتين: رسالة للبائع تحتوي على معلومات طلبية الشراء OI (Order Information) ، ورسالة للبنك تحتوي على معلومات الدفع PI (Payment Information)، ويقع توليد بصمتي رسالتة الطلبية OIMD (OI Message Digest) والدفع PIMD (PI Message Digest). ثم يقع توقيع الجميع لنحصل على توقيع ثنائي(Dual Signature).

- عملية توليد طلبية شراء من طرف العميل:

يقوم العميل بتشفير معلومات الدفع مع التشفير الثنائي وبصمة طلبية الشراء بمفتاح الجلسة K_s. ثم يشفر المفتاح K_s بالمفتاح العام للبنك KU_b لتكوين الظرف الرقمي. تدمج هذه النواتج مع بصمة معلومات الدفع، وطلبية

الشراء، والتوقيع الثنائي، وشهادة حامل بطاقة الدفع في رسالة واحدة وترسل إلى البائع (انظر إلى الشكل 22.4).

شكل 22.4 طلبية شراء باستعمال SET.

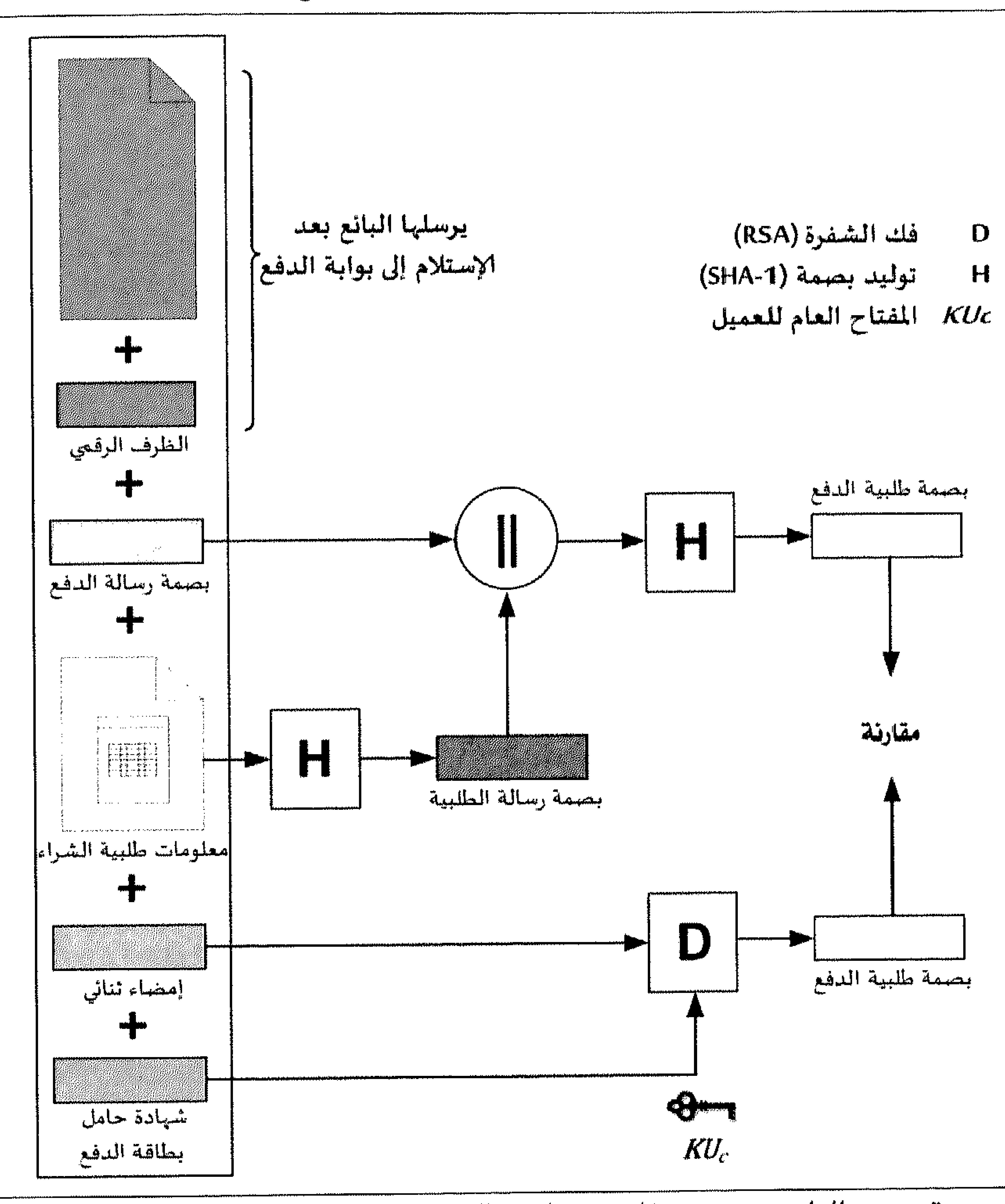


- عملية معالجة طلبية شراء من طرف البائع:

يقوم البائع بفك تشفير التوقيع الثنائي باستعمال المفتاح العام للعميل الذي يتم استخراجه من الشهادة، ويحصل بذلك على بصمة طلبية الدفع. يقوم البائع أيضًا بحساب بصمة طلبية العميل، ثم يستخلص طلب الدفع باستعمال بصمة معلومات الدفع ليقوم بحساب بصمة طلبية الدفع من

جديد. بعد ذلك يقوم بمقارنة بصمتي طلبية الدفع، للتأكد بأن الطلبية ومعلومات الدفع لم تتغير (انظر إلى الشكل 4. 23).

شكل 4. 23 معالجة طلبية شراء.



يقوم البائع بعد ذلك بإرسال معلومات البيع لبوابة الدفع PaymentGateway للحصول على إذن بالسحب، ويرسل رسالة قبول لطلبية الشراء للعميل.

تقوم بوابة الدفع بما يلي:

1. التحقق من كل الشهادات.
2. فك شفرة الظرف الرقمي للإذن بالدفع لاستخلاص مفتاح الجلسة، ومن ثم فك شفرة الإذن بالدفع.
3. التحقق من توقيع البائع على الإذن بالدفع.
4. فك شفرة الظرف الرقمي لمعلومات الدفع لاستخلاص مفتاح الجلسة، ومن ثم فك شفرة معلومات الدفع.
5. التحقق من التوقيع الثنائي لمعلومات الدفع.
6. التحقق من أن رقم المعاملة المستقبل من البائع مساو لرقم المعاملة المستخلصة من معلومات الدفع المستقبلة من العميل.
7. طلب واستقبال الإذن بالدفع من مصدر الشهادات.
8. إرسال إذن بإكمال عملية البيع للبائع.

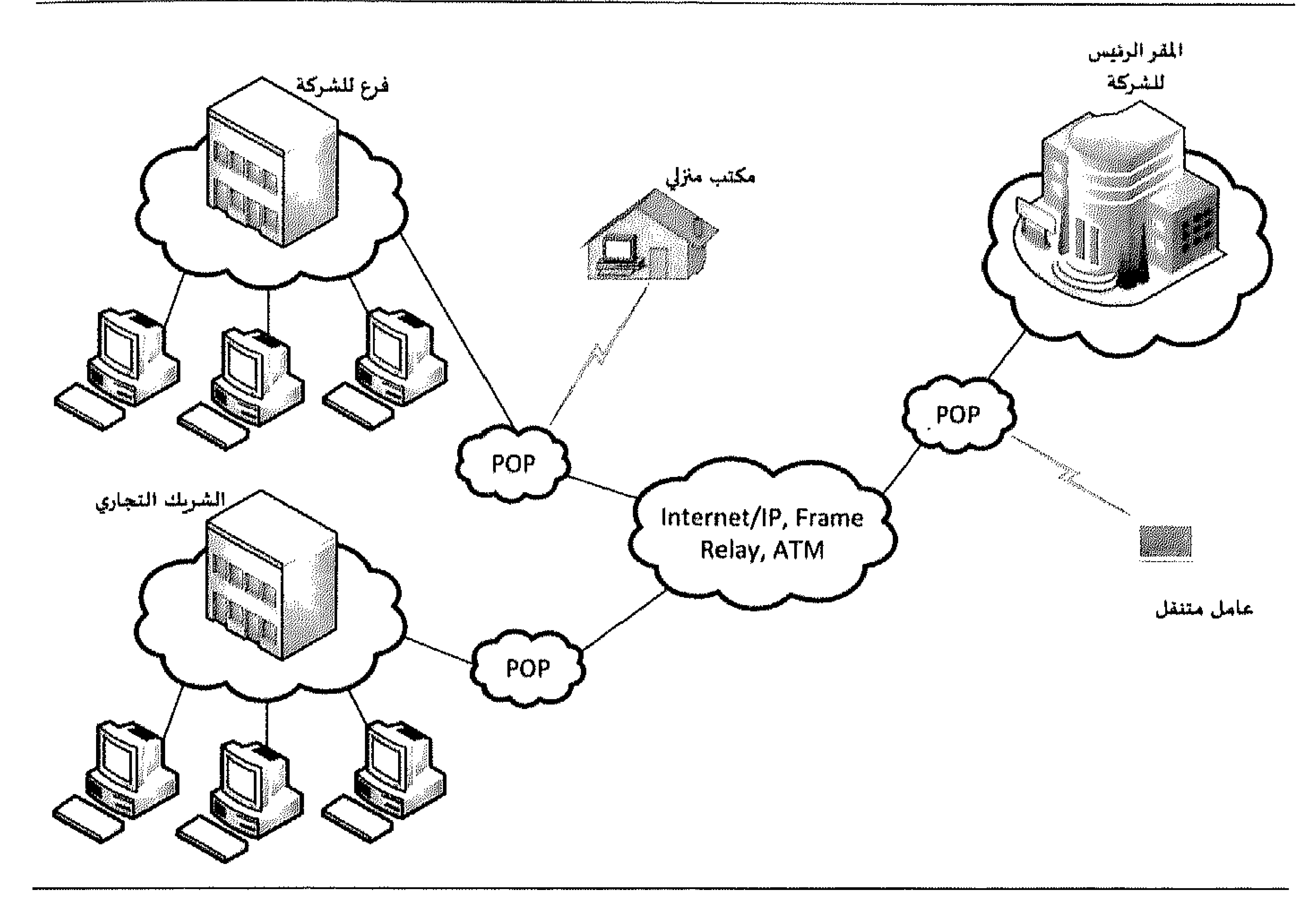
يقوم البائع بعدها بإرسال طلبية قبض أو سحب المبلغ المالي لبوابة الدفع التي تقوم من التحقق من صحة الطلبية، ثم تقوم بتحويل المبلغ إلى حساب البائع، وتشعره بإتمام العملية بإرسال سند القبض.

3. 5 الشبكة الافتراضية الخاصة

قبل ظهور الشبكات العامة – وعلى رأسها الإنترنت – كان هناك ما يعرف بالشبكات الخاصة التي تعتمد في ربطها على الخطوط المستأجرة leased lines، وذلك لتحقيق الاتصال الآمن، لكن هذا النوع من الاتصال يكلف الكثير، ويحتاج إلى تجهيزات كالموديم. وبالمقابل فإن الإنترنت واستخدامها كوسيط لربط الشبكات الخاصة غير مكلفة، لكنها غير آمنة. فظهرت فكرة استحداث شبكات خاصة تتم عبر الانترنت ولها قدر عالٍ من الأمان، وهذا ما يعرف بالشبكات الافتراضية الخاصة Virtual Private Network (VPN). وبهذه التقنية تمكنت الشركات من التخلص من كلفة الخطوط المؤجرة، وتكاليف نقل المعلومات الخاصة بها بين فروعها البعيدة، وبين المقر الرئيس لها، مع ضمان سرية البيانات وسلامتها. كما ساهمت هذه التقنية في توفير الوقت على المستخدم في

الوصول إلى معلوماته المتوفرة في جهازه عن بعد في أي مكان وزمان، والتعامل مع الشريك التجاري بشكل موثوق (انظر إلى الشكل 23.4).

شكل 4. 24الشبكات الافتراضية الخاصة.



وعموماً فإن لهذه الشبكات فوائد عدة، أهمها:

1. جعل المعلومة متاحة في أي مكان وزمان.
2. الاتصال عن بعد يجعل المعلومة سهلة وآنية التبادل.
3. جعل المستخدم البعيد لا يشعر بالعزلة.

عمليا، تعتبر الشبكات الخاصة الافتراضية بمثابة برمجيات يشغلها طرفا لاتصال على الأجهزة المستخدمة في الاتصال بينهما عبر الإنترنت، وتبادل البيانات بشكل آمن وسري باستعمال تقنيات التشفير. تعمل هذه برمجيات كمرشح رزم من حيث سماحها للبيانات بالانتقال من جهاز إلى جهاز آخر فقط، تم إعدادهما بشكل جيد للتعامل مع VPN؛ إذ إن أي خلل في إنشاء الشبة الافتراضية سوف يعرض المعلومات للخطر. تقوم هذه البرمجيات بتشفير البيانات لضمان سرية البيانات عند عبورها عبر الإنترنت.

هناك ثلاثة اختيارات يمكن المفاضلة بينها عند تحديد كيفية تركيب الشبكات الخاصة الافتراضية VPN هي:

1. تركيب شبكة تعتمد على جدار الحماية الناري.
2. تركيب شبكة تعتمد على الموجه.
3. تركيب شبكة تستخدم برمجيات، وأجهزة متخصصة لهذا الغرض.

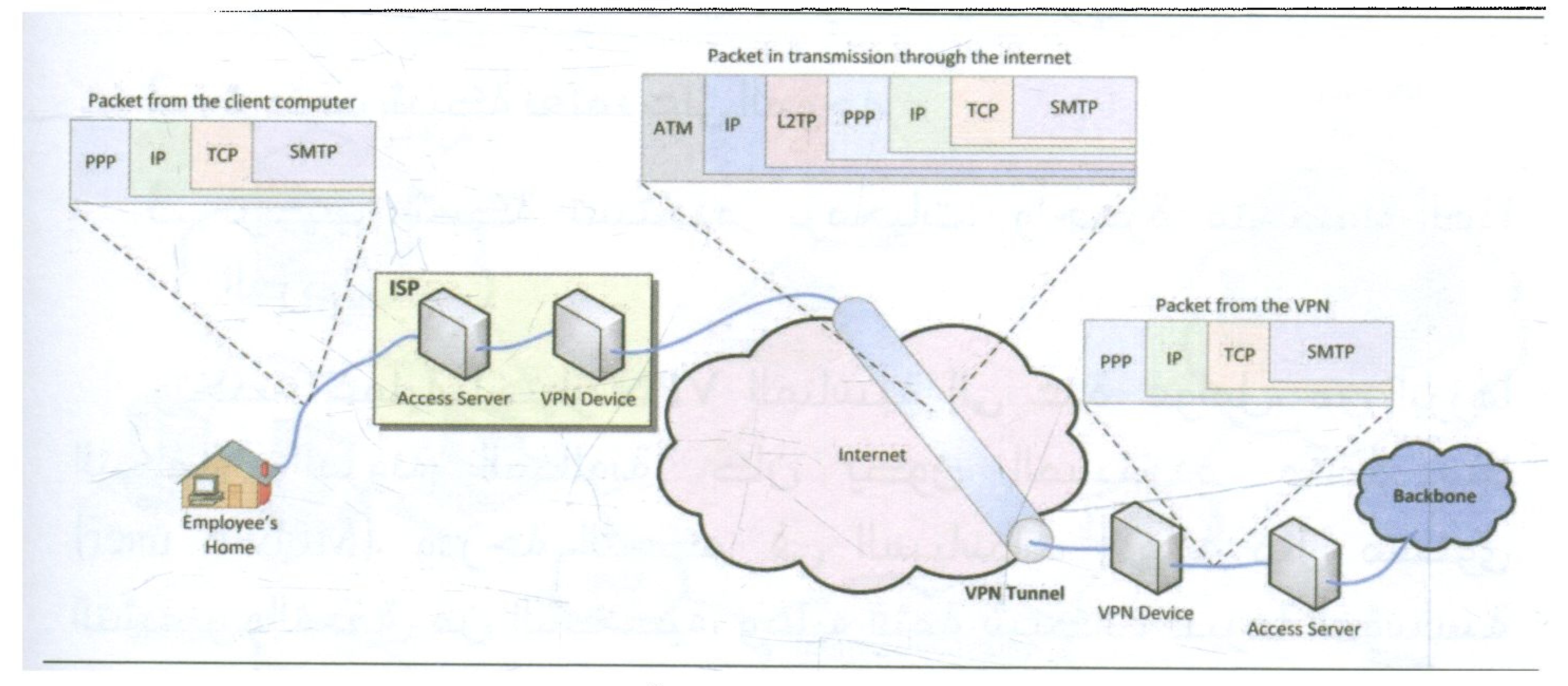
وتخضع عملية اختيار VPN المناسبة إلى عدة عوامل، من أبرزها التكلفة والمرونة المطلوبة، كأن يكون المستخدم متنقلا مثلا (Mobile user)، ودرجة التحكم في السياسات الأمنية مثل مستوى التشفير والتحقق من الشخصية؛ وعليه فثمة شبكات تديرها المؤسسة بالكامل مما يتيح للمؤسسة التحكم بشبكتها وسياستها الأمنية. وثمة شبكات يديرها مزود الخدمة لتحقيق خدمات آمنة بين بعض الشركات وعملائها عبر الانترنت، مثل أن يشترط مصرف ضمان سرية وسلامة معاملات العميل على حسابه عندما يستعمل الإنترنت.

بروتوكولات الشبكات الافتراضية:

تندرج الشبكة الافتراضية الخاصة في الطبقتين الثانية والثالثة في معيار OSI للشبكات. وتستعمل تقنية الأنفاق التي تمكن المرسل من تغليف بياناته في طرود برتوكول الإنترنت IP حيث تخفي بنية التوجيه والتحويل لشبكة الإنترنت لضمان عدم المساس بها من طرف الدخلاء. نستعمل أساسًا ثلاثة بروتوكولات في تنفيذ تقنية الأنفاق، وهي: برتوكول الإنترنت الآمن IPSec وبرتوكولPPTP:point to point tunneling protocol الذي يعتمد عل برتوكول PPP الذي يستعمل شبكات الهاتف الثابتة، وأخيراً برتوكول تقنية النفق في الطبقة الثانية L2TP: Layer 2 Tunneling Protocol المؤلف من برتوكول PPTP وبرتوكول إعادة توجيه الطبقة الثانية L2F، انظر إلى الشكل الموالي حيث يقع تغليف الرسالة التي يرسلها المستخدم باستعمال برتوكول PPP ليقع تغليفه ببروتوكول

L2TP ثم ببرتوكول IP ثم ATM وهكذا نكون قد أنشأنا نفقًا – أي شبكة افتراضية – باستخدام برتوكول L2TP.

شكل 4. 25 تقنية النفق باستعمال L2TP



يقع استعمال تقنية الأنفاق لإنشاء أنواعٍ مختلفة من الشبكات الافتراضية الخاصة وهي أساسًا ثلاثة أنواع:

1. شبكة الولوج عن بعد Remote Access VPN: وهي خاصة بالموظفين المتنقلين حيث تتيح لهم الدخول على مواردهم في شبكة الشركة من أي مكان، وفي أي وقت ما عن طريق اتصال مباشر على الإنترنت، أو عن طريق شبكة الهاتف الثابت، وذلك عن طريق إنشاء نفق باستعمال بروتوكولات VPN.

2. الشبكة الافتراضية للشبكة الداخلية Intranet VPN: وهي الشبكة التي تربط الفرع الرئيس للمؤسسة بالفروع الأخرى البعيدة من خلال الإنترنت.

3. الشبكة الافتراضية للشبكة الخارجية Extranet VPN: وهي الشبكة التي تربط المؤسسة بالأطراف الخارجية التي تتعامل معها، كالشريك التجاري، أو مكاتب الاستشارات وغيرها.

6 - أهم مصطلحات الفصل

Firewall	جدار حماية
Packet Filter	مصفي طرود
Proxy or Application Gateway	خادم وكيل أو بوابة تطبيقية
Firewall policy	سياسة جدار الحماية
Filtering Rule	قاعدة تصفية
Screened host	جهاز فحص متعدد البطاقات الشبكية
(Demilitarized Zone (DMZ	منطقة منزوعة السلاح أو المنطقة الخضراء
Socket	مقبس
Port	منفذ
Bastion host	جهاز المعقل الحصين
Honeypots	جهاز جرة العسل أو الشرك المغري
Honeynets	شبكة العسل أو الشبكة المفخخة
(Intrusion Detection System (IDS	نظام اكتشاف الاختراقات
(Demilitarized zone (DMZ	المنطقة الخضراء
Anomaly based IDS	مراقبة سلوك الشبكة
Misuse IDS	مراقبة سلامة الطرود
Hybrid IDS	نظم اكتشاف الاختراقات الهجينة
Alert	انذار
Digital pests	الأوبئة الرقمية
Dormant phase	طور السكون
Propagation phase	طور الانتشار
Triggering phase	طور الإثارة وقدح الزناد

Execution phase	طور التشغيل
Trojan Horse	حصان طروادة
Virus	الفيروسات
Bacterium	البكتيريا
Worms	الدودة
Trapdoor	الباب الخلفي
Logic bomb	القنبلة المنطقية
Parasitic	الفيروسات الطفيلية
Virus checker	فاحص الفيروسات
Email security	أمن البريد الإلكتروني
Web security	أمن الويب
Change Cipher specific protocol	بروتوكول تغيير التشفير
Alert Protocol	برتوكول الإشعار
Handshake Protocol	برتوكول المصافحة
SET	برتوكول العمليات التجارية الآمنة
Virtual Private Network	الشبكة الافتراضية الخاصة
Remote Access VPN	شبكة الولوج عن بعد
Intranet VPN	الشبكة الافتراضية للشبكة الداخلية
Extranet VPN	الشبكة الافتراضية للشبكة الخارجية

7 - تمارين الفصل

1. ما الخيار الذي لا يعتبر بندًا في السياسة الأمنية لجدران الحماية؟
   - أ. الاستعمال المقبول أو المرضي.
   - ب. تجهيز التقارير.
   - ت. الاتصال بالشبكة.
   - ث. المستخدمين المتعاقدين.

2. ما الطريقتان التي تستعمل عادة في تصميم الأمن في الشبكات؟
   - أ. مصفي الطرود وخادم البيانات.
   - ب. مصفي الطرود والخادم الوكيل.
   - ت. الانترنت والخادم الوكيل.

3. ما الخيار الأول والذي كان الأكثر شيوعًا في جدران الحماية لحماية الشبكة؟
   - أ. مصفي الطرود.
   - ب. البوابة التطبيقية.
   - ت. الخادم الوكيل.

4. المقبس هو عنوان انترنت:
   - أ. مع عنوان اثرنت.
   - ب. مع رقم منفذ.
   - ت. مع مصفي طرود.
   - ث. مع خادم وكيل.

5. ما الخيار الذي يعبر الشكل الأقدم في مصفي الطرود؟
   - أ. التصفية عن طريق عنوان الإنترنت.
   - ب. التصفية عن طريق منفذ TCP ومنفذ UDP.
   - ت. التصفية على حسب نوع البرتوكول.
   - ث. التصفية على حسب نوع تجزئة الطرد.

6. ما الخيار الذي لا يمثل ميزة ذات فائدة للخادم الوكيل؟
   أ. إخفاء العملاء.
   ب. تصفية المحتوى.
   ت. خادم وكيل لكل خدمة.
   ث. سجل تتبع واحد.
7. المعقل الحصين هو:
   أ. جهاز مصمم لاستقطاب المهاجمين.
   ب. جهاز محصن أمنيًّا من أي جهاز آخر في الشبكة.
   ت. منطقة أنشئت لتمكين مستخدمي الإنترنت من الوصول للخدمات للشبكة الداخلية.
8. لا بد من اعتبار المسائل القانونية عند إنشاء جرة عسل:
   أ. نعم.
   ب. لا.
9. ما الحل الأفضل عندما يتعرض جهاز المعقل الحصين للهجوم؟
   أ. استعادته عن طريق نسخة احتياطية.
   ب. تهيئته وتثبيته من جديد.
   ت. تحديث كلمات سر المدراء.
10. من المفروض أن نضع جرة العسل:
   أ. في خادم الويب.
   ب. في المنطقة الخضراء DMZ.
   ت. في الشبكة الداخلية.
11. ما الخيار لحذف سلسلة من IPTABLES
   D–
   X–
   F–
   L–

12. ما الخيار لإضافة قاعدة في IPTABLES

A–

N–

D–

s–

13. في أي طبقات الأوزي يعمل جدار الحماية.

1و 3و 5و 7

2 و3 و4 و7

2 و3 و4 و6

2 و3 و5 و7

14. من مهام الجدار الناري:

1. تركيز الإجراءات الأمنية في نقطة واحدة.

2. تسجيل وقائع الاستخدام، وكافة المعلومات عند حركة المرور به.

3. الحد من تعرض الشبكة الداخلية لأخطار الاتصال بالشبكات العامة.

4. جميع ما ذكر.

15. ما نوع الهجوم الفيروسي الحاسوبي الذي يغير إشارته وترتيب أوامره، أو حتى استخدام دوال لها نفس الوظيفة، وذلك في كل مرة يقوم فيها بنسخ نفسه في جهاز آخر ولماذا؟

16. إذا كان البرنامج التالي مخزنًا في ذاكرة جهاز الحاسب الآلي كما هو مبين أسفل، وذلك ابتداءً من العنوان H0801 إلى العنوان 1000 من الذاكرة، مع العلم أن العناوين من H 1001 إلى H 1102 خالية من أي أمر أو معلومة، كيف يتم إقحام الفيروس التالي نفسه داخل هذا البرنامج بحيث يبتدئا من العنوان H0900 من الذاكرة بين ذلك بتعبئة الحيز location من H1001 إلى H 1102 من الذاكرة بالأوامر المناسبة:

أوامر الفيروس:

```
VInstruction 1 (start)
..................
..................
..................
VInstruction 100 (end)
```

البرنامج المهاجم:

Memory	Address
	0000
. . .	
PInstruction-1	0801
PInstruction-2	0802
. . .	
PInstruction-m	0900
PInstruction-m+1	0901
. . .	
PInstruction-n	1000
	1001
. . .	
	1100
	1101
	1102
. .	

Memory	Address
	0000
. . .	
PInstruction-1	0801
PInstruction-2	0802
. . .	
....	0900
PInstruction-m+1	0901
. . .	
PInstruction-n	1000
....	1001
. . .	
....	1100
....	1101
....	1102
. .	

الْفَصْلُ الْخَامِس

سِيَاسَاتُ أَمْنِ الْمَعْلُومَات

يَهْدِفُ هَذَا الْفَصْلُ إِلَى:

1. التعريف بأهمية وأهداف سياسات أمن المعلومات، كأداة مهمة من أدوات أمن المعلومات.
2. التعريف بأنواع سياسات أمن المعلومات.
3. إعطاء نبذة عن كيفية إعداد سياسات أمن المعلومات.
4. إعطاء نبذة عن أهم المعايير القياسية، وطرق تحليل المخاطر العالمية.

1 - مقدمة الفصل

تقوم السياسات بشكل عام برسم الخطوط العريضة التي تحدد القواعد واللوائح، أو الضوابط التقنية والمادية، التي تحكم كيفية إعداد النظام وتشغيله، كما تحكم أيضًا تصرفات الموظفين، وطبيعة عملهم داخل المنظمة تجاه مجال ما، وذلك في الحالات الاعتيادية، أو الحالات الطارئة، حيث يجب أن تحتوي سياسات أمن المعلومات على قسم يشرح الإجراءات الواجب اتخاذها في حال حدوث كارثة، وبالتالي تحديد كيفية وتوقيت استرجاع البيانات عند حدوث هجوم يؤدي لضررها، وتحديد كيفية صد أي هجوم، وتحديد كيفية ومكان حفظ النسخ الاحتياطية، وتعيين المسئولين عليها. على ذلك فإن سياسات أمن المعلومات تؤدي وظيفتين أساسيتين هما:

الأولى: تحديد الإجراءات الأمنية الواجب الأخذ بها لتحقق الأمن داخل المنظمة، عن طريق قواعد تحكم كيفية تهيئة النظام.

الثانية: تحكم تصرفات وأعمال الموظفين بالمنشأة، وذلك بوضع قواعد تشرح ما هو متوقع من الموظف أن يقوم به، وما هو غير مسموح القيام به لدى التعامل مع التقنية، ومع المعلومات داخل المنشأة، وبذلك فهي تجعل العاملين في المنظمة شركاء في تحقيق الأمن المعلوماتي.

2 - أهداف سياسات امن المعلومات

تسعى سياسات أمن المعلومات إلى عدد من الأهداف منها:

تعريف المستخدمين والإداريين بالتزاماتهم، وواجباتهم المطلوبة تجاه حماية المعلومات وأنظمتها وشبكاتها بكافة أشكالها، وفي مراحل إدخالها ومعالجتها وتخزينها، ونقلها وإعادة استرجاعها.

كما تهدف سياسات أمن المعلومات إلى تحديد الإجراءات الإلكترونية التي يتم من خلالها تحقيق وتنفيذ الواجبات المحددة على كل من له علاقة بالمعلومات، ونظمها وتحديد المسؤوليات عند حصول الخطر.

بيان الإجراءات المتبعة لتجاوز التهديدات والمخاطر والتعامل معها، وتحديد الجهات والأشخاص المناط بها القيام بذلك.

3 - أساس سياسات أمن المعلومات

تعتمد سياسات أمن المعلومات من عدد من الأسس، منها: تحديد المخاطر، وأغراض الحماية، ومواطن الحماية، وأنماط الحماية اللازمة، وإجراءات الوقاية من المخاطر، وتتلخص تلك الأسس التي تبنى عليها سياسات أمن المعلومات القائمة على الاحتياجات المتباينة لكل منشأة من الإجابة عن تساؤلات ثلاثة رئيسة وهي: ماذا أريد أن أحمي؟، من ماذا أحمي المعلومات؟، كيف أحمي المعلومات؟

4 - أنواع سياسات أمن المعلومات

هناك عدد من السياسات التي يجب الأخذ بها عند صياغة سياسات أمن المعلومات، ومن أهم هذه السياسات سياسات المعلومات، وسياسات الأمن، وسياسات استخدام أجهزة الحاسب، وسياسات استخدام الإنترنت، وسياسات التعامل مع البريد الإلكتروني. فكل من هذه السياسات تعتمد على عدد من الركائز المشتركة، وهي:

الغرض: لا بد أن يكون هناك تحديد واضح للغرض من ووضع السياسات والإجراءات وبيان فوائدها.

نطاق العمل: كل مجموعة من السياسيات والإجراءات لا بد أن تحدد البيئة التي سوف تطبق فيها، فمثلا سياسات الأمن قابلة للتطبيق على جميع أنظمة الحاسب وشبكاته بينما سياسات المعلومات فنطاق تطبيقها هو البيانات.

المسؤولية: تعرّف وتحدد من الذي سوف يكون مسئولاً في كل إدارة أو قسم في المنظمة عن الالتزام بتطبيق كل جزئية في السياسات. إن كل شخص في المنشأة يستخدم نظام المعلومات وشبكاته، هو شخص مسئول عن الأمن فيها. تنقسم المسؤولية إلى قسمين:

1. **المركز الرئيس:** الذي يحتوي على النظام الرئيسي للمؤسسة أي السيرفر وفيه يتم تحديد الصلاحيات للمستخدمين، إن هذه الأنظمة عادة ما تكون ذات مواصفات عالية ويتم عليها حفظ بيانات المؤسسة، والقيام بالمعالجات اللازمة.

2. **أنظمة الاتصال والشبكة:** وهي المسئولة عن ربط أجهزة المستخدمين معًا، وتأمين الاتصال، وتبادل المعلومات ضمن فروع المؤسسة داخليًا، وبين فروعها والمحيط الخارجي أو مع شبكة خارجية.

4.1 سياسات المعلومات

لا تتساوى كافة الأصول في جميع نظم المعلومات، من حيث أهميتها على صعيدي العمليات وتحقيق رؤية الجهة ومهمتها، فأهمية بعض الأصول تفوق أهمية الأصول الأخرى، ومن هنا فإنها بحاجة إلى عناية وحماية إضافيتين، مما يوجب تصنيف الأصول المعلوماتية طبقًا لذلك.

وستؤدي عملية التصنيف إلى تمكين المنشأة من تركيز آليات الحماية على تلك الأصول التي تكون عرضة أكثر من غيرها لمخاطر محددة بسبب قيمتها العالية. وسيتم تصنيف الأصول المعلوماتية طبقًا لاحتمالات تعرضها للمخاطر.

تحدد سياسات المعلومات درجة حساسية (سرية) المعلومة بهدف تحديد مستوى الإجراء الأمني لها، طبقًا لمستوى سريتها، وحساسيتها، وقيمتها، وحيويتها، إذاً لا بد من وجود سياسات خاصة في التعامل مع المعلومات من حيث تحديد أهميتها، فمثلا هناك معلومات تحتاج إلى مستوى عالٍ من الأمن، كما أن هناك معلومات تحتاج إلى مستوى أقل من الأمن وهكذا، فهناك عدة أمور يجب مراعاتها عند وضع سياسات المعلومات وهي كالتالي:

1. **تحديد المعلومات الحساسة:** مثل سجل الأعمال، وتصميم المنتج، ومعلومات المريض الصحية. فسجل الإعمال مثل الخطة التسويقية، ومعلومات العملاء وغيره تعتبر من خصوصيات المنظمة، لا ينبغي أن يطلع عليها سوى المسئولين فيها، وكذلك المعلومات الخاصة بكيفية تصنيع منتج ما يجب ألا يطلع عليه المنافسون، وعلى المستوى الفردي فإن المعلومات الخاصة بإصابة مريض بمرض خطير تعتبر من المعلومات ذات الحساسية العالية، التي قد تؤثر على المريض نفسه لو تم كشفها لجهات أخرى كشركات التامين. مع العلم إنه ليس جميع المعلومات لها صفة الحساسية طوال الوقت، أي قد لا تكون كل

معلومة لها صفة السرية أو الخصوصية بشكل دائم، لذلك لا بد لها من مراجعة بشكل مستمر وبين فترة وأخرى، فمعلومات المريض الصحية قد لا تصبح سرية عند وفاة المريض أو شفائه.

2. تصنيف المعلومات

يمكن تصنيف المعلومات من حيث حساسيتها وسريتها ما بين اثنين إلى ثلاث تصنيفات وذلك على النحو التالي:

- معلومات عامة، وهي المعلومات التي تكون معروضة للاطلاع عليها من قبل أي شخص، وهذه لا تحتاج إلى حماية ضد الكشف والاطلاع على المعلومة.
- معلومات تخص المنظمة وموظفيها وعملائها (لا تكشف للعامة)، وهذه تحتاج إلى نظام وصول يخول موظفي وعملاء المنظمة فقط.
- معلومات حساسة وهذه تعتبر ذات درجة عالية من الحساسية، يجب معها تكثيف الأمن عليها لكي لا يطلع عليها إلا من مقبل أشخاص محددين.

3. وضع علامات تحديد مستوى الحساسية:

من الأمور المهمة عند العمل على سياسات المعلومات تحديد كيفية وضع علامات تبين مدى حساسية وسرية المعلومة وأهميتها، مثل وضع علامة تبين أن هذه المعلومة سرية أو سرية للغاية إلى آخره.

4. تخزين المعلومات الحساسة:

تتولى سياسات المعلومات تحديد المستوى المطلوب لحماية المعلومات، وهذا يتم باستخدام أنظمة التحكم بالوصول للمعلومة، واستخدام اسم المستخدم وكلمة السر، كما أنه في حال المعلومات عالية الحساسية قد نضطر إلى تشفيرها قبل أن تحفظ في وسائط التخزين.

5. تراسل المعلومات الحساسة:

تحدد سياسات المعلومات الكيفية التي يتم بها تراسل المعلومات الحساسة، إما عبر البريد، أو الفاكس، أو البريد الإلكتروني أو غيره، كما تحدد مدى حاجة المعلومات المراد تراسلها إلى عملية تشفير، أي أن طرق إرسال المعلومات يتوقف على مدى حساسيتها وأهميتها.

6. التخلص من المعلومات الحساسة

وأخيرا فهناك أمر مهم قد يتم تغافله، وهو كيفية التخلص من المعلومات الحساسة، فلا بد من أن تحدد سياسات المعلومات الطريقة المناسبة التي يجب إتباعها عند الرغبة في التخلص من المعلومات، فمثلا عند إجراء تعديل بسيط في أسئلة الامتحانات النهائية للطلاب فلا بد من إتلاف النسخة القديمة منها فوراً، وبطريقة آمنة، وذلك باستخدام برامج متقدمة بدلا من استخدام أسلوب الحذف المعتاد الذي يؤدي بالمحذوف إلى سلة المهملات.

2. 4 سياسات الأمن

تعنى السياسات الأمنية بتحدد متطلبات الأمن التقنية العامة بناء على سياسات المعلومات المقرة وذلك لجعل نظام المعلومات وشبكاته أكثر أمناً، وبذلك تقوم بتحديد كيفية تهيئة النظام من الناحية الأمنية والمواصفات ولاختبارات الأمنية المطلوبة للأنظمة المراد استخدامها. هناك عدة آليات يجب مراعاتها في سياسات الأمن منها ما يلي:

1. **آلية تحديد الهوية والتحقق: منها** تقوم السياسات الأمنية بتحديد كيفية إعطاء للمستخدم وذلك على ضوء معيار محدد، كما أنها في المقابل تحدد آلية العمل التحقق من هوية المستخدم سوى كان مستخدمًا عاديًا أو مسئولاً ذا صلاحيات أعلى، أي أن سياسات الأمن تعنى بالتأكد من الهوية بكيفية اختبار المستخدمين فيما إذا أنهم فعلا المستخدمون الحقيقيون, إن طرق التحقق من الهوية تتراوح من وجود رقم معرف و كلمة سر خاصة بكل مستخدم إلى طرق التأكد التي تعتمد على معدات مادية، مثل التأكد عن طريق بطاقات ممغنطة، أو التأكد من البصمة وحدقة العين، وطبعًا يتراوح استخدام هذه التقنيات حسب الأهمية والحاجة.
2. **آلية التحكم في الوصول:** السياسات الأمنية لا بد أن تحدد المعايير الفنية المطلوبة لعمل أنظمة الوصول للملفات، أي إنها يهتم بتحديد هوية من يدخل الشبكة، وإلى أين يدخل وما سبب دخوله، حيث يجب أن توجد إجراءات واضحة للتأكد من أن الأشخاص الحقيقيين هم

الذين يمتلكون حق الوصول للخدمات والمعلومات الخاصة بهم دون غيرهم، وبالتالي يجب أن تكون السياسة قادرة على منح المديرين المرونة الكافية لمنح الصلاحيات للمستخدمين، والتحكم بها منعاً لحدوث أخطاء.

3. **آلية التدقيق والمراجعة:** ما إن توضع السياسات الأمنية لنظام معلومات، ويتم تنفيذها إلا ويتوجب بعد فترة التأكد من أن المكونات والموظفين يطبقون ويلتزمون هذه السياسة، ويتم ذلك بمراقبة تطبيق هذه السياسة, تفيد السياسة في التعرف على المشاكل، والتنبؤ بها قبل حدوثها ويجب مراجعة السياسة باستمرار للتأكد من استمرار فعاليتها. كما أنه ومن خلال سياسة التدقيق يتم تحديد أنواع وأشكال الأعمال التي تحتاج إلى عمل تدقيق لها على جميع الأنظمة في المنشأة.
4. **الارتباط بالشبكة:** لا بد أن تحدد السياسات الأمنية القواعد الضرورية؛ لربط النظم بالشبكة وآليات ومتطلبات الحماية بها، وجود أدوات قياس ومراقبة للوصول، مثل وجود نظام الجدار الناري، وأنظمة رصد الاختراقات، وأجهزة مراقبة الشبكة، إضافة إلى تحديد بعض الخدمات، مثل خدمات الوصول عن بعد, خدمات مشاركة الملفات. ومن حيث الدخول إلى المعلومات عبر الشبكة فلا بد من سياسات دخول واضحة تحدد حقوق وامتيازات كل شخص في المنشأة للوصول إلى ملفات أو مواقع معينة في النظام، إضافة إلى سياسة بشأن التعامل مع الاتصالات الخارجية، وأجهزة ووسائل الاتصال المستخدمة، إضافة البرامج إلى الشبكة الجديدة.
5. **التشفير وإدارة المفتاح:** يعتبر التشفير أحد الأمور المهمة لتحقيق حماية المعطيات المرسلة عبر الشبكة من الكشف والاطلاع عليها، وكذلك من التعديل والتزوير، خاصة في ظل وجود شبكة إنترانت، كما تحدد السياسات الأمنية مستوى التشفير المطلوب، والخوارزمية المناسب لذلك. إن كل عملية تشفير تتطلب وجود مفتاح، ولذلك يجب أن تحدد السياسات الأمنية عدة أمور مهمة عند اختيار المفتاح، مثل طول المفتاح، ومدة صلاحيته، وآلية توليد المفتاح، وكيفية توزيع

المفاتيح، إضافة إلى تحديد نظام البنية التحتية للمفاتيح العامة والشهادات الرقمية لحماية الخدمات الإلكترونية.

4.3 سياسات استخدام الحاسب الآلي

تحدد القوانين والقواعد التي تعين من المسموح له باستخدام النظام وكيفية الاستخدام مثلاً:

- لدخول النظام هل يستخدم كلمة مرور مباشرة.
- التطبيق المخول للموظف العمل ضمنه، مثل موظف المبيعات يعمل على نظام المبيعات.
- استخدام الأجهزة لعمل المنظمة فقط، ولا يجوز استخدامها لأغراض شخصية.
- تحديد أوقات عمل الأجهزة، والوقت والفترة الزمنية يفترض أن يستخدم النظام من قبل الموظف.
- نسخ المستندات من الأجهزة إلى أقراص وإخراجها خارج المنظمة، وتحديد النسخ الدوري (يومي،أسبوعي،شهري) لجميع أنظمة المعلومات.
- المراجعة الدورية لسجلات النظام.

4.4 سياسات استخدام الانترنت

تبين سياسات استخدام الانترنت إمكانية وطريقة استخدام الموظفين للإنترنت، وتحديد وقت استخدام الانترنت، إضافة إلى سياسة حجب لمواقع على الشبكة العكبوتية، كما تحدد إمكانية تنزيل البرامج وتحديد نوعيتها على السيرفر، وهل هي تجارية أو غير تجارية، وهل هي موثوقة أم غير موثوقة، بالإضافة لتقييد تحميل البرامج من الإنترنت وتنصيبها على السيرفر مباشرة.

4.5 سياسات استخدام البريد الإلكتروني

يوضع هذا النوع من السياسات لتنظيم وتقيتين استخدام البريد الإلكتروني في المنشأة مثل الأتي:

- عدم فتح البريد الإلكتروني قبل فحصه وضمان خلوه من الفيروسات.

- عدم استخدامه في الأمور الخاصة.
- الاتفاقات والعقود المراسلة عبر البريد الإلكتروني لا تعتبر عقودا رسمية، لكي لا تتورط المنظمة في ذلك إلا إذا نص على ذلك.
- تذيل أي رسالة تخرج عبر نظام البريد الإلكتروني للمنشأة بعبارة تبين إخلاء مسئولية المنظمة عن أي مشكلة تحصل من استخدام البريد الإلكتروني، وذلك لكي لا يستغل من قبل الموظفين حتى لا تتعرض المنظمة للمسئولية مستقبلاً.

5 - إعداد سياسات امن المعلومات

لدى إعداد أية سياسة بشأن أمن المعلومات، ولكي تكون هذه السياسة فاعلة ومنتجة وهادفة، لا بد أن يساهم في إعدادها وتفهمها وتقبلها وتنفيذها مختلف مستويات الوظيفة في المنشأة الواحدة، إضافة إلى حاجتها إلى التعاون والدعم الكامل من الكافة، من هنا فإن المعنيين بإعداد سياسة أمن المعلومات يتوزعون إلى مراتب وجهات عديدة داخل المنشأة، لكن بوجه عام يشمل مسئولي أمن المعلومات، ومديري الشبكات، وموظفي وحدة الكمبيوتر، ومديري الوحدات المختلفة في المنشأة، وتشمل أيضًا فريق الاستجابة للحوادث والأعطال، وممثلي مجموعات المستخدمين، ومستويات الإدارة العليا إلى جانب الإدارة القانونية.

5.1 وثيقة أمن المعلومات

إذا قبل الشروع في كتابة وثيقة سياسة أمن المعلومات فإنه يتوجب تحديد العناصر التالية بشكل دقيق:

1. **الهدف:** يجب عند التفكير في كتابة سياسات الأمنية بالتفكير أن يكون لدينا فكرة واضحة عن أهداف هذه السياسة.
2. **المدى:** وهو ما ستقوم بحمايته بواسطة سياستك الأمنية، وهذا يشمل ذكر كل المجالات اللازمة للحماية، انطلاقًا من الحماية الفيزيائية حتى الشخصية، ومدى شمولية هذه السياسة من مدراء ومستخدمين، وحتى زوار.

3. **اعتماد الوثيقة من قبل الإدارة العليا:** بعد معرفة أهداف السياسة، وتحديد نطاقها ومداها، يجب إطلاع الإدارة العليا للمنشأة عليها، وأخذ الموافقة الخطية، ومناقشة طرق تحقيق هذه السياسة.
4. **الاطلاع على سياسات أخرى:** قبل الشروع بكتابة السياسة الأمنية الخاصة بمنشأة ما يجب أن يتم دراسة سياسات عامة، وتجارب منشآت أخرى للمجال، والاطلاع على المعايير العلمية في هذا المجال.
5. **تقدير المخاطر:** قبل كتابة السياسة يجب عليك تحديد المخاطر المتوقعة، والأساليب المتاحة لمواجهتها.
6. **تحديد مكونات السياسة ومن ثم كتابتها:** يعتمد ذلك على دراسة المخاطرة، وليس من الضروري وضع كل المكونات السابقة عند كتابة سياسة أمنية.
7. **نشر الوثيقة السياسات:** العمل على تعريف العاملين في المنشأة بهذه السياسات، كلٌّ فيما يخصه مع التأكيد على توضيح أهمية هذه السياسة.
8. **التقييم:** بعد كتابة السياسة ينبغي القيام بتقييمها، وذلك بالإجابة عن بعض الأسئلة التي تساعد في تقييم ما تم كتابته من سياسات، ومن أمثلة هذه الأسئلة ما يلي: "هل تتوافق سياستك مع القانون؟"، "هل السياسات تقيد نفوذ الموظفين؟"، "هل هي قابلة للتطبيق العملي؟"، "ما مدى توافقها مع المعايير الدولية مثل معايير ISO BS7799 أو النسخة الثانية "(ISO/ICE 27001)؟

5.2 سياسات أمن المعلومات الناجعة

من حيث فعالية الاستخدام لكي توصف سياسات أمن المعلومات بأنها سياسات ناجحة، يتعين أن تعمم بشكل شامل على كافة قطاعات الإدارة وأن تكون مقبولة وواقعية، ومن المناط به تنفيذها إلى جانب توفر الأدلة التوجيهية والإرشادية، لضمان إدامة التنفيذ، وعدم التقاعس فيه، والتنفيذ هنا هو الاستخدام الفعلي لأدوات الحماية التقنية من جهة، والتطبيق الفعلي لقواعد العمل والتعامل مع البيانات ونظمها من جهة أخرى، ولا تحقق السياسات الأمنية نجاحًا – إن كان ثمة غموض فيها –،

لهذا لا بد أن تكون واضحة ودقيقة في محتواها، ومفهومة لدى كافة المعنيين.

أما من حيث المحتوى فإن سياسات أمن المعلومات تمتد إلى العديد من المناحي المتصلة بنظم المعلومات وإدارتها والتعامل معها، إضافة إلى المسائل المتعلقة بالمعلومات ذاتها وتعامل الغير مع معلومات المنشأة، وكذلك اقتناء وشراء الأجهزة التقنية وأدواتها، والبرمجيات، والحلول المتصلة بالعمل، والحلول المتعلقة بإدارة النظام. كما تبين الاستثناءات التي تعتمدها السياسات على حق الخصوصية لموظفي المنشأة، مع مبررات هذه الاستثناءات، كرقابة البريد الإلكتروني مثلا، أو رقابة الدخول إلى المنشأة، أو رقابة الوصول إلى ملفات المستخدمين بالمنشأة.

وتضم سياسات المعلومات أيضًا قواعد الاشتراكات التي تحدد سياسة المنشأة بشأن اشتراكات الغير في شبكتها أو نظمها، وكذلك سياسات التعامل مع المخاطر والأخطاء، بحيث تحدد ماهية المخاطر، وإجراءات إبلاغ عنها، والتعامل معها، والجهات المسئولة عن التعامل مع هذه المخاطر.

وفي كل الحالات لضمان جودة ونجاعة سياسات أمن المعلومات، لا بد من استعمال طريقة من الطرق العالمية في تحليل المخاطر، والتأكد من توافقها مع المعايير القياسية العالمية.

6 - المعايير القياسية العالمية وطرق تحليل المخاطر

6.1 أبرز الهيئات والمعايير المختصة في الأمن المعلوماتي

ظهرت الحاجة إلى ما يسمى بإدارة نظام أمن المعلومات، وتبنت بعض الهيئات الأكاديمية، والجمعيات المهنية المتخصصة، كالاتحاد العالمي للشهادات الاحترافية في أمن المعلومات (ICS2)، وجمعية تدقيق وضبط أنظمة المعلومات (ISACA)، ومنظمة المواصفات والمقاييس العالمية (ISO)، والهيئة الإلكتروتكنولوجية الدولية IEC، ومؤسسة NIST الأمريكية والبريطانية، وبعض المؤسسات التي تجري بعض الإحصائيات والأرقام في بعض دول العالم مثل منظمة CLUSIF الفرنسية المختصة في السياسات الأمنية، ونظيرتها منظمة CISSA الأمريكية، كما أنه يوجد بعض المؤسسات المتخصصة في خدمات الأعمال والرقابة والاستشارات

مثل مؤسسةERNST & YOUNG التي تعد من المؤسسات القائدة في هذا الحقل في 130 دولة في العالم، إضافة إلى بعض المكاتب المتابعة للأمن المعلوماتي، وفرق عمل من أشهرها فريق سيرت CERT، وهو فريق فدرالي للتدخل السريع بشأن الجريمة الإلكترونية.

وتعتبر شهادة المعيار العالمي لأمن المعلومات ISO27001 من أهم الشهادات المعترف بها عالمياً، والتي تحمل قيمة عالية من ناحية رفع الكفاءة والجودة في المنظمات حول العالم في مجال إدارة نظم المعلومات.

كما أن معيار ISO/IEC 27002 هي أحد أفراد عائلة معايير ISO/IECISMS المتكاثرة باستمرار، وهي المعايير الخاصة بحماية المعلومات التي نشرتها المؤسسة الدولية للمعايير، ISO، والهيئة الإلكتروتكنولوجية الدولية، IEC، تحت رقم ISO/IEC 17799:2005، ثم أعادت ترقيمها في يوليو 2007؛ ISO/IEC 27002:2005، لكي تتماثل مع سواها من سلاسل معايير ISO/IEC 27000. وهي تحمل اسم: تكنولوجيا المعلومات – تقنيات الأمن والحماية – الإجراءات المثالية لإدارة شؤون حماية المعلومات. النسخة الحالية من المعايير هي نسخة معدلة من الإصدار الأول الذي أصدرته ISO/IEC في عام 2000، والتي كانت بدورها صورة طبق الأصل من المعايير البريطانية رقم BS 7799-1:1999.كما طورت هيئة المعايير البريطانية معايير أخرى مثل FIPS31, FIPS140, FIPS188, BS7799-2. كما طورت شركات بطاقات الائتمان الكبرى؛ لتساعد الشركات والمؤسسات التي تتعامل مع الدفع بواسطة بطاقة الائتمان على تجنب حالات الاحتيال بواسطة بطاقات الائتمان، والحصول على المعلومات بطريقة غير شرعية، وغير ذلك من التهديدات والثغرات الأمنية معيار PCIDSS. يجب على الشركة التي تتعامل مع بطاقات الائتمان أو تخزن بياناتها أو ترسلها أن تتوافق مع معايير PCIDSS، والشركات غير المتوافقة التي تتعامل مع واحدة أو أكثر من بطاقات الائتمان، بصورة مباشرة أو عبر وسيط، مهددة بأن تخسر قدرتها على التعامل مع بطاقات الائتمان، أو محاسبتها أو تغريمها.

على كل إدارة تتبع ISO/IEC 27000 أن تقوم بتحديد موجوداتها المعلوماتية من قواعد بيانات ومستندات، إلى البرامج والأجهزة المعلوماتية وتوابعها، وبعدها يأتي تصنيف تلك الموجودات بحسب أهميتها من الناحية الأمنية، بغية معرفة درجات الوقاية والحماية اللازمة للمحافظة عليها، ولهذه الغاية وجب ترقيم الموجودات، وتحديد كيفية معالجة كل صنف منها، وذلك خلال النسخ والحفظ وإرسال المعلومات بالبريد العادي أو بالبريد الإلكتروني، بالإضافة إلى عملية تلف المعلومات.

في معيار الأيزو ISO/IEC 27002:2005 يؤكد على أي مؤسسة أن تحدد حاجياتها الأمنية بالاعتماد على ثلاثة مصادر رئيسة: (1) تحليل المخاطر (2) الحاجيات النظامية القانونية والعلاقات مع الأطراف الأخرى (3) مبادءها وأهدافها من نظام معالجة معلوماتها. إن معيار الأيزو ISO/IEC 27002:2005 يقدم جملة من التوصيات، ولكن لا يقترح طريقة لإنشاء نظام إدارة الأمن بالكامل. بينما معيار ISO/IEC 27001 يوفر نموذجا لهذه الإدارة. فهو يؤكد على ضرورة استعمال طرق تحليل المخاطر دون أن ينص على طريقة بعينها. يعتمد هذا المعيار على نموذج يتكون من أربع عمليات تكرر دوريًا وهي (التخطيط، التنفيذ، المراقبة، التعديل)، وهي عملية دورية، فبعد التخطيط للعمليات الأمنية يقع تنفيذها، ثم مراقبة أدائها، وفي حالة الخطأ يقع التدخل بالتعديل المناسب، ثم إعادة التخطيط بناء على ما حدث، وهكذا دواليك. كما يقدم المعيار جملة من نقاط المراقبة في ملحقاته، ويمكن المؤسسات من التأكد من موافقة نظم حمايتها لحاجياتها الأمنية، وسيرها على حسب تخطيطها.

6 . 2 أبرز طرق تحليل المخاطر

يصف المعيار الدولي الجديد ISO/IEC 27005:2008 عملية إدارة معلومات أمن المخاطر، والمهام المتعلقة بها لمساعدة المؤسسات في إدارة المخاطر. فيقرر أنه قد تكون التهديدات عرضية أو متعمدة، وقد يكون لها علاقة باستخدام أو تطبيق نظم تكنولوجيا المعلومات، أو النواحي البيئية والفيزيائية التابعة لكتنولوجيا المعلومات. قد تأخذ هذه المخاطر أي شكل من أشكال سرقة المعلومات، أو مخاطر متابعة الأعمال عن طريق

الانترنت، أو التجسس عن بعد، أو سرقة المعدات أو الوثائق من خلال أي ظاهرة مناخية، كالزلازل أو الحرائق أو الفيضانات أو الحوادث الوبائية. قد ينتج عن هذه المخاطر العديد من الآثار السلبية على العمل، مثل الخسارة المالية أو الضرر المادي أو ضياع خدمات الشبكات الرئيسة، أو خسارة ثقة المستهلك، نتيجة لفقدان إمداد الطاقة، أو إخفاق معدات الاتصالات.
ويعرف التهديد: أنه مجموعة مؤلفة من عدة متتابعات ناجمة عن وقوع حادث غير مرغوب فيه، أو الاحتمال القوي لوقوع حادث ما. ويصف تقييم المخاطر نوعية وكمية التهديد، وبهذا يمكن تحديد أولوية المخاطر تبعًا لإدراك جديتها، أو غيرها من المعايير الموضوعة. كما يقدم معيار ISO/IEC 27005:2008 إرشادات حول إدارة مخاطر أمن المعلومات، والتي تستند إلى طريقة إدارة المخاطر، معرفة المفاهيم والنماذج والعمليات والمصطلحات التي وضعها هذا المعيار. وتتألف العمليات من: 1- تقييم المخاطر. 2- علاج المخاطر. 3-قبول المخاطر. 4-ترابط المخاطر. 5- مراقبة المخاطر. غير أن المعيار المذكور لا يقدم أية معلومات محددة منهجيه لإدارة المخاطر الأمنية. والأمر متروك للمؤسسة لتحديد نهج لإدارة المخاطر بما يتوافق مع تخصصها.

من جهة أخرى ظهرت عدة طرق لتقييم وإدارة المخاطر، منها: طرق مجانية كـEbios و Austrian IT Secuity Handbook و Dutch A&K analysis و IT-Grundschutz و Octave و SP800-30، وكلها تصلح لكل أنواع المؤسسات ما عدا Octave فإنها للمؤسسات الصغيرة الحجم. بعض هذه الطرق لا تشترط لاستعمالها ترخيص، وبعضها يستوجب تراخيص مجانية. كما طورت لبعض هذه الطرق أدوات تسهل استعمالها لكنها في المجمل تتطلب خبرة في التعامل معها لاستعمالها الاستعمال الأمثل. بالمقابل هناك طرق أخرى غير مجانية مثل Cramm و ISF methods والآيزو بكافة معاييره، وMeharis التي طورت لها أداة Risicare وهذه الطرق تصلح لكل أحجام المؤسسات ما عدا ISF methods فإنها لا تصلح للمؤسسات الصغيرة الحجم، وكل هذه الطرق تتطلب خبرة للتعامل معها.

جدول 1.5 طرق تحليل المخاطر.

	تحديد التهديد	توصيف التهديد	تقييم درجة التعرض	توصيف الخطر	تقييم الخطر	معالجة الخطر	درجة قابلية الخطر	اتصالية الخطر	اللغة
Ebios	***	***	***	***	***	***	***	***	انجليزية، فرنسية، ألمانية، إسبانية
Austrian IT Secuity Handbook	**	*	*	**	***	***	***	***	ألمانية
Dutch A&K analysis	***	***	***	***					هولندية
IT-Grundschutz	***	***	***	***	***	***	***	***	انجليزية، ألمانية
Octave	**	**	**	**	**	**	**	**	انجليزية
SP800-30	***	***		***	***	***	***		انجليزية
Cramm	***	***	***	***					انجليزية، هولندية، تشيكية
ISF methods	***	***	***	***	***	***	***	***	انجليزية
Mehari	***	***	***	***					انجليزية، فرنسية
ISO/ICE 27005	**	**	**	**	**	***	***	***	انجليزية

يعرض الجدول 1 هذه الطرق ومدى تقدمها في خصائص وعمليات تقييم وإدارة المخاطر، بحيث تشير(***، ** ، *) إلى درجة استكمال هذه الخاصية. ويمكن تقسيم هذه الطرق إلى أربعة أنواع: (قياسي مثل ISF وISO)، وعام مثل الطرق المعتمدة على BS7799، ومفصل مثل CRAMM وMEHARI وهاتان الأخيرتان يمكن تنفيذها آليا، نظرًا لقابليتها للبرمجة، وأخيرًا مهجن، وهي الطرق التي تدمج الطرق العامة والمفصلة.

7- خاتمة

يعد تبني السياسة الأمنية الصحيحة والشاملة بمثابة حجر الأساس لأمن تقنية المعلومات، إذ تفيد دراسة حديثة أن60% من الاختراقات يمكن تفاديها باتخاذ سياسة أمنية فعالة، ولضمان فعالية أمن تقنية المعلومات يتوجب القيام بتوثيق السياسة الأمنية، وإطلاع المستخدمين عليها، كما يجب أن تخضع هذه السياسة للمراجعة الدورية مع مراقبة مستويات الالتزام بها، فضلاً عن تكريس الدعم لها من أعلى مستويات الإدارة، وبهذا يتم تقييم السياسة الأمنية خلال فترات معينة، وبعدها يمكن الاستفادة من الثغرات – إن وجدت –، وإعادة تعديل السياسة لرفع كفاءتها. إن تطوير سياسة أمنية فعالة ليس بالمهمة السهلة على الإطلاق، فالموجودات والأصول التي قد تحتاج إلى حماية تتضمن المعلومات، والبرامج والتطبيقات، والأجهزة والتسهيلات، إذ إن تحديد مستوى الحماية المطلوبة مثلاً في عالم الشركات يأخذ بعين الاعتبار متطلبات الشريك التجاري، وتوقعات المستهلكين، والتعليمات الحكومية، والمعايير الدولية المتطورة مثل ISO17799، إضافة إلى تحديد معايير الأمن اللازمة، والممارسات الصحيحة. ومن هذا المنطلق فإن بناء وتطوير نظام آلٍي مساعد في توصيف وبناء المنظومة الأمنية الكاملة بدءا من تقييم وتوصيف السياسات الأمنية باستعمال طرق تحليل المخاطر إلى تخطيط، وتصميم عمارة البنية التحتية الأمنية للمنشآت، يعد من أهم الأمور التي تعني المنشآت لرأب الصدع بين سياساتها، والمعايير القياسية الدولية، والمساعدة في تحديد الاستثمار الأمني المطلوب / وتطبيقه بكلفة مجدية.

8 - أهم مصطلحات الفصل

Security policies	السياسات الأمنية
Policy goals	أهداف السياسة
Policy Scope	نطاق العمل
Accountability	المسؤولية
Policy publishing	نشر السياسة
Sensitive data specification	تحديد المعلومات الحساسة
Information classification	تصنيف المعلومات
Sensitive rate	مستوى الحساسية
Authentication mechanism	آلية تحديد الهوية والتحقق
Access control mechanism	آلية التحكم في الوصول
Verification and reviewing mechanism	آلية التدقيق والمراجعة
Cryptography and Key management	التشفير وإدارة المفتاح
Computer policy	سياسات استخدام الحاسب الآلي
Internet policy	سياسات استخدام الانترنت
Email policy	سياسات استخدام البريد الإلكتروني
Information security agreement	وثيقة أمن المعلومات
Risk evaluation	تقييم المخاطر
threat	التهديد
Risk management	إدارة المخاطر
Risk analysis	تحليل المخاطر
Crisis management	إدارة الأزمات

المصادر والمراجع

- William Stallings: Cryptography and Network Security - Principles and Practice, 4th edition, Prentice Hall, 2004.
- McClure, S., Scambray, J., and Kurtz, G. Hacking Exposed: Network Security Secrets & Solutions, 4th ed. McGraw-Hill. February 2003.
- C. Kaufman, R. Perlman, and M. Speciner, Network Security: Private Communication in a Public World, second ed. Prentice Hall, 2002.
- W. Stallings, Network Security Essentials --- Applications and Standards, Prentice-Hall, Englewood Cliffs, NJ, U.S.A., 2000.
- Andrew S. Tanenbaum. Network security. In Computer Networks, pages 577-620, 1996 by Prentice Hall PTR.
- SCHNEIER, B. Applied cryptography: protocols, algorithms, and sourcecode in C. John Wiley and Sons, New York, 1996.
- J. Porto, H. Krumm and P.L. de Geus, "Policy Modeling and Refinement for Network Security Systems", 6th IEEE International Workshop on Policies for Distributed Systems and Networks, pp. 24--33, 2005.
- GEER, D. Malicious Bots Threaten Network Security. IEEE Computer 38, 1 (Jan. 2005), 18--20.
- Joshua D. Guttman and Amy L. Herzog. Rigorous automated network security management. International Journal for Information Security, 2004. Forthcoming.
- Bruce Schneier. Hacking the business climate for network security. IEEE Computer, pages 87--89, April 2004.
- A. Zhou, J. Blustein, and N. Zincir-Heywood, "The state of network security management: Issues and directions," Dalhousie University Faculty of Computer Science, Technical Report CS-2003.
- J. G. Goodall, A. Komlodi, and W. G. Lutters. Information visualization for intrusion detection analysis: A needs assessment of systems and network

security experts. In Workshop on Statistical and Machine Learning Techniques in Computer Intrusion Detection, Fairfax, VA, 2003.

- SysAdmin, Audit, Network, Security Institute. The Twenty Most Critical Internet Security Vulnerabilities(Updated) -- The Experts Consensus. 08 Oct. 2003. Version 4.0. The SANS Institute. http://www.sans.org/top20/top20.pdf
- G. Vigna. Teaching network security through live exercises. In C. E. Irvine and H. L. Armstrong, editors, World Conference on Information Security Education, volume 253 of IFIP Conference Proceedings, pages 3--18. Kluwer, 2003.
- Austin, Brice, Dinehard et al. "Basic Security Policy" 1.2 SANS Security Essentials II: Network Security Overview. SANS 2003. pp2-20.
- Tieyan Li, Wu Yongdong. "Trust on Web Browser: Attack vs. Defense". International Conference on Applied Cryptography and Network Security (ACNS'03). Kunming China. Oct. 16-19, 2003. Springer LNCS.
- D. R. Stinson, Cryptography: Theory and practice, CRC Press, Boca Raton, FL, 2002.
- Mike D. Schiman, "Building Open Source Network Security Tools". Wiley 2003.
- Schneier, B. Managed Security Monitoring: Network security for the 21st century. Computers and Security, pp. 491--503, vol. 20, num. 6, 2001.
- Network Security: Principles and Protocol Standards, Tutorial T352, Network World+Interop, 2001, Stephen Kent, GTE Networking.
- Avolio, Frederic M. "Best Practices in Network Security". 20 March 2000. URL: http://www.networkcomputing.com/1105/1105f2.html
- Alfred J. Menezes, Paul C. Van Oorschot, and Scott A. Vanstone. Handbook of Applied Cryptography. CRC Press, Boca Raton, FL, 1997.
- D.F. Ferraiolo, R. Sandhu, S. Gavrila, D.R. Kuhn and R. Chandramouli. Proposed NIST Standard for RoleBased Access Control. ACM Transactions on Information and System Security, vol. 4, pp. 224-274, 2001.

- Fred B. Schneider, Enforceable Security Policies, ACM Transactions on Information and System Security 3,1 (2000) p. 30-50.
- D. Eastlake, Domain Name System Security Extensions. IETF - Network Working Group, March 1999. RFC2535.
- S. N. Chari and P.-C. Cheng. Bluebox: A policy-driven, host-based intrusion detection system. In Proceedings of the Network and Distributed System Security Symposium, 2002.
- J.A. Goguen and J. Meseguer, "Security Policies and Security Models," Proc. 1982 IEEE Symp. Security and Privacy, IEEE Press, 1982, pp. 11--20.
- D. Clark and D. Wilson. A Comparison of Commercial and Military Computer Security Policies. In Proc. of the IEEE Symposium on Security and Privacy, April 1987.
- H. H. Hamed, E. S. Al-Shaer, and W. Marrero. Modeling and verification of ipsec and vpn security policies. In ICNP, pages 259--278, 2005. 4, 15, 18
- F. Martins and V. Vasconcelos. Controlling security policies in a distributed environment. DI/FCUL TR 04--1, Department of Informatics, University of Lisbon, April 2004.
- Stephen Chong and Andrew C. Myers. Security policies for downgrading. In Proceedings of the ACM conference on Computer and communications security, pages 198--209, New York, NY, USA, 2004. ACM Press.
- Seo, Y.-W., Giampapa, J., and Sycara, K., A multi-agent system for enforcing Need-To-Know security policies, In Proceedings of International Conference on Autonomous Agents and Multi Agent Systems (AAMAS) Workshop on Agent Oriented Information Systems (AOIS-04), pp. 163-179, 2004.
- Ryutov, T. and Neuman, C. The Specification and Enforcement of Advanced Security Policies. In Third International Workshop on Policies for Distributed Systems and Networks (POLICY 2002.
- Markus Schumacher, " Security Patterns and Security Standards - With Selected Security Patterns for Anonymity and Privacy," European Conference on Pattern Languages of Programs (EuroPLoP), 2002.

- Humphreys, E.J. (Editor), Taxonomy for Security Standardization - Part 1 - Introduction and Overview of Security Standards, CEN/CENELEC, CSecG/49/90, September 1990
- Hafner, M., Breu, R., Breu, M.: A security architecture for inter-organizational workflows: Putting security standards for web services together. In Chen, C.S., et al., J.F., eds.: Proceedings ICEIS. (2005)
- CESG.: CESG COMPUSEC Memorandum No 10 - Minimum Computer Security Standards for HMG Information Handled by Information Technology Systems, Issue 2.2, (October 1996).
- Kjell Hausken. Returns to Information Security Investment: The Effectof Alternative Information Security Breach Functions on OptimalInvestment and Sensitivity to Vulnerability. Information Systems Frontiers, 5(8), 2006.
- Farkas, C. Web and Information Security, chapter Data Confidentiality on The Semantic Web: Is There an InferenceProblem ? Chapter IV, pp. 73--91. Idea Group Inc, 2006.
- Rae, A. J. & Fidge, C. J. (2005a), 'Identifying critical components during information security evaluations ', Journal of Research and Practice in Information Technology 37(4), 391--402.
- John Wilander and Jens Gustavsson. Security requirements---a field study of current practice. In E-Proceedings of the Symposium on Requirements Engineering for Information Security, in conjunction with the 13th IEEE International Requirements Engineering Conference, Paris, France, http://www.sreis.org, August 2005.
- A. Jsang, D. Bradley, and S.J. Knapskog. Belief-Based Risk Analysis. In Proceedings of the Australasian Information Security Workshop (AISW), Dunedin, January 2004.
- Wood CC. Why information security is now multi-disciplinary, multi-departmental, and multi-organizational in nature. Comput Fraud Secur 2004.

- Karin Hone and J. H. P. Eloff, Information Security Policy – what do international information security standards say, Technical Report, Department of Computer science, Rand Afrikaans University, 2002.
- Malcolm E. Palmer, Craig Robinson, Jody C. Patilla, and Edward P. Moser, Information Policy Framework: Best Practices for Security Policy in the E-Commerce Age, Information systems Security, V. 10 , No. 2 pp 13-22, May 2001.
- Mohammed Alabdulkareem, Information Security in Saudi Arabia, 2nd. Saudi IT Security Forum 2005.
- Electronic Crime Needs, Assessment for State and Local Law Enforcement. U.S. Department of Justice, National Institute of Justice, National Institute of Justice, March 2001.
- Arab-British Chamber of commerce (A-BCC), Computer hackers to be given ahard time, science & technology, Vol 8, No 3, Marck 1992. P.5.
- Arab-British Chamber of commerce (A-BCC), Computer hackers to be given ahard time, science & technology, Vol 8, No 3, Marck 1992. P.5.
- Electronic Crime Needs, Assessment for State and Local Law Enforcement. U.S. Department of Justice, March 2001.
- Security Barometer Survey document. The Psychology of Security www.quocirca.com
- Handbook of Information Security Management, Hal Tipton and Micki Krause, Consulting Editors, NIST 2003.
- Distributed Systems concepts and Design, couloirs, Dollimore, Kingberg, Addison-Wesley, edition 4, 2006.
- كتاب سياسات أمن المعلومات (Information Security Policies) تأليف: د. محمد عبد الله القاسم 1426 هـ – 2005م
- قاموس مفردات أمن المعلومات، خالد بن سليمان الغثبر، سراج الدين أحمد امبابي، مركز التميز لأمن المعلومات، 2010م

- أمن المعلومات بلغة ميسرة، خالد بن سليمان الغثبر، محمد بن عبدالله القحطاني، مركز التميز لأمن المعلومات، 2010م

- الاصطياد الالكتروني، خالد بن سليمان الغثبر ،سليمان بن عبد العزيز بن هيشة، مركز التميز لأمن المعلومات، 2010م

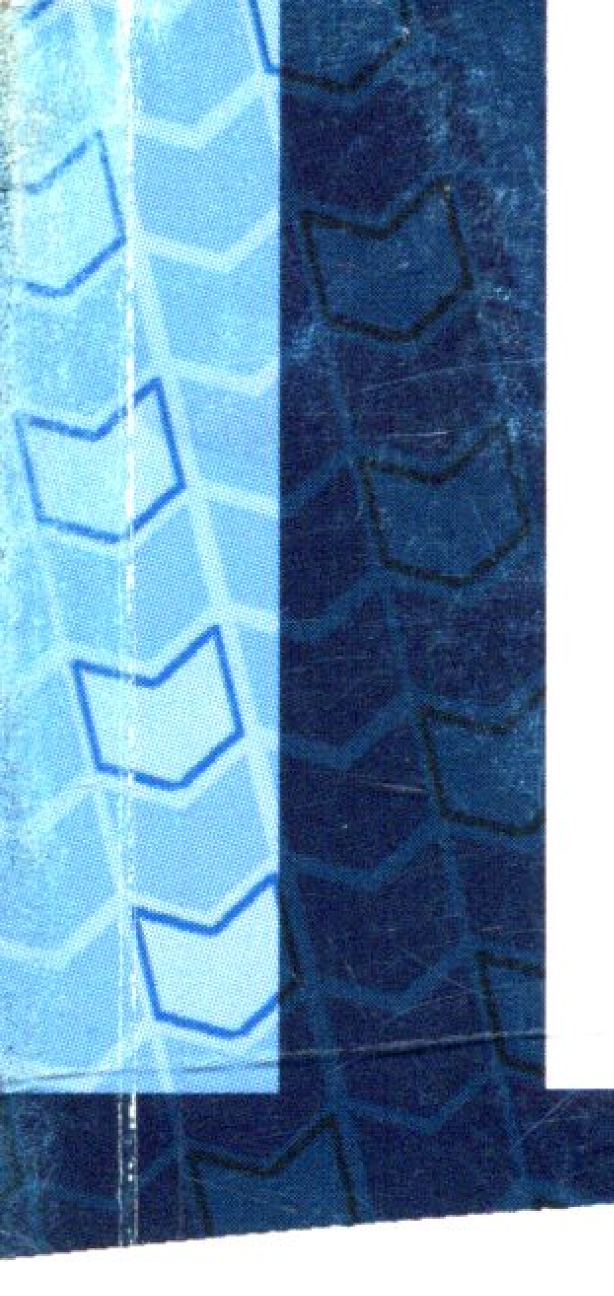
- الأستاذ (Ulrich Seiber) جرائم الحاسب الآلي والجرائم الأخرى المرتبطة بالتقنيات الحديثة لوسائل الاتصال، ترجمة الدكتور سامي الشوا، ورقة عمل مقدمة للمؤتمر السادس للجمعية المصرية للقانون الجنائي، 25–28، تشرين أول /أكتوبر 1993

قائمة الأشكال

قائمة الجداول

المصطلحات

الفهرس



ردمك : ٩٧٨-٦٠٣-٥٠٥-٢٠٩-٢

www.imamu.edu.sa
e-mail: journal@imamu.edu.sa